

testowanie i dokumentowanie aplikacji

tutor version 1.6

# Software bugs

Software bugs in the [Therac-25](#) radiation therapy machine were directly responsible for patient deaths in the 1980s.

In 1994, an [RAF Chinook helicopter crashed](#), killing 29; was initially blamed on pilot error, but was later thought to have been caused by a software bug in the [engine-control computer](#).

Buggy software caused the early 21st century [British Post Office scandal](#)

In 1996, the European Space Agency's US\$1 billion prototype [Ariane 5 rocket was destroyed](#) less than a minute after launch due to a bug in the on-board guidance computer program

# ISTQB

ISTQB® to wiodący światowy system certyfikacji w dziedzinie testowania oprogramowania



About Us ▾

Certifications

Training & Exams ▾

🔍 What would you like to search?

News SCR

## Welcome to ISTQB®

**ISTQB® is the leading global certification scheme in the field of software testing.**

As of June 2023, ISTQB® has administered **1.3 million** exams and issued more than **957k** certifications in over **130 countries**. With its extensive network of [Accredited Training Providers](#), [Member Boards](#), and [Exam Providers](#), ISTQB® is one of the biggest and most established vendor-neutral professional certification schemes in the world. ISTQB® terminology is industry recognized as the defacto language in the field of software testing and connects professionals worldwide.

Join our network of software testing professionals by becoming certified today!

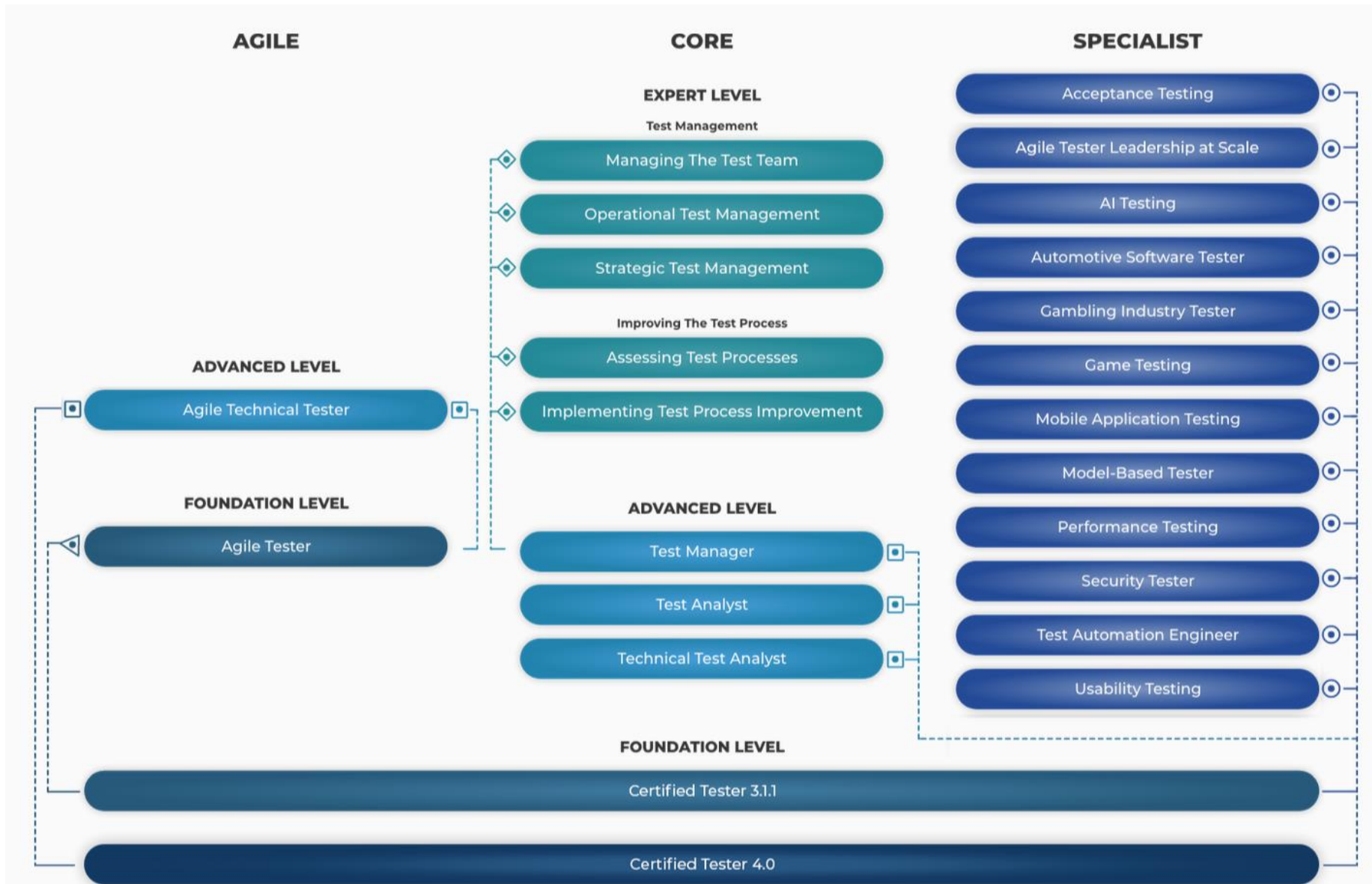
Our certifications

Work With Us

<https://www.istqb.org/>

# ISTQB

zestaw certyfikatów oferowanych przez ISTQB



<https://www.istqb.org/>

# ISTQB

## Słownik terminów testowych ISTQB oraz QUIZ

The screenshot shows the ISTQB Glossary website interface. At the top left is the ISTQB Glossary logo. Next to it is a search bar with the placeholder text 'Wyszukaj...'. To the right of the search bar is a blue circular icon with a white 'H' symbol. Further right is a navigation menu with the following items: 'WYSZUKAJ' (highlighted with a blue underline), 'QUIZ', 'HISTORIA', 'POMOC', 'GRAFY', and 'POLISH' with a dropdown arrow. Below the navigation menu, there is a message: 'Kliknij termin, aby wyświetlić więcej informacji na jego temat'. On the right side of this message, it says '585 Znalezione wyniki' next to a red circular icon with a white printer symbol. Below this is a section titled 'Wszystkie terminy'. The first term listed is 'adekwatność funkcjonalna' with the description: 'Stopień, w jakim funkcje ułatwiają realizację określonych zadań i celów.' The second term is 'analityczna strategia testowa' with the description: 'Strategia testowa, podczas której zespół testowy analizuje podstawę testów, by zidentyfikować te warunki testowe, które należy pokryć.' The third term is 'analiza drzewa usterek' with the description: 'Metoda używana do analizy przyczyn usterek (defektów). Technika modeluje wizualnie to, jak związki logiczne pomiędzy awariami, błędami człowieka i zewnętrznymi zdarzeniami mogą powodować powstawanie specyficznych defektów. (FTA - akronim od ang. Fault Tree Analysis).' The fourth term is 'analiza dynamiczna' with the description: 'Proces oceny modułu lub systemu oparty na jego zachowaniu podczas wykonania.' The fifth term is 'analiza przepływu danych' with the description: 'Rodzaj analizy statycznej przeprowadzanej w oparciu o cykl życia zmiennych.'

[https://glossary.istqb.org/pl/PL/search?term=&exact\\_matches\\_first=true](https://glossary.istqb.org/pl/PL/search?term=&exact_matches_first=true)

# ISTQB



materiały do pobrania – Sylabus, przykładowe pytania egzaminacyjne

[O NAS](#) [AKTUALNOŚCI](#) [NASZE INICJATYWY](#) [PUBLIKACJE](#) [ISTQB®](#) [IREB®](#) [TMMI®](#) [A4Q](#)

## ISTQB® Do pobrania

WSTECZ

Polskie materiały pomocne w przygotowaniu do egzaminu ISTQB®: sylabusy oraz egzaminy przykładowe odnoszące się do różnych poziomów certyfikacji testerskich: Certyfikowany Tester Poziom Podstawowy, certyfikacje na Poziomie Zaawansowanym, certyfikacje na poziomie Specjalista oraz ze ścieżki Agile (Poziom Podstawowy i Zaawansowany) dostępne są poniżej Słownika.

[ISTQB® Certyfikowany Tester - Poziom Podstawowy w. 4.0 - Sylabus wersja angielska](#)

<https://sjsi.org/ist-qb/do-pobrania/>

## testowanie oprogramowania

Testowanie  
oprogramowania  
polega na sprawdzeniu,  
czy oprogramowanie  
spełnia oczekiwania.

[https://en.wikipedia.org/wiki/Software\\_testing](https://en.wikipedia.org/wiki/Software_testing)



## testowanie oprogramowania ISTQB

Testowanie oprogramowania to zbiór czynności mających na celu wykrycie defektów i dokonanie oceny jakości artefaktów związanych z oprogramowaniem.



# zasady testowania ISTQB

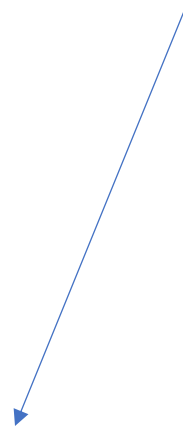
- **Testowanie ujawnia defekty, ale nie może dowieść ich braku.**
- **Testowanie gruntowne jest niemożliwe**
- **Wczesne testowanie oszczędza czas i pieniądze**
- **Defekty mogą się kumulować**
- **Testy ulegają zużyciu**
- **Testowanie zależy od kontekstu**
- **Przekonanie o braku defektów jest błędem**

# Czynności i zadania testowe ISTQB

- 1. Planowanie testów**
- 2. Monitorowanie testów i nadzór nad testami**
- 3. Analiza testów**
- 4. Projektowanie testów**
- 5. Implementacja testów**
- 6. Wykonywanie testów**
- 7. Ukończenie testów**

# Czynności i zadania testowe ISTQB

**Stworzyć dokumentację testową w oparciu o materiały dostępne na poniższej stronie**



<https://sii.pl/blog/produkty-procesu-testowego-od-podstawy-do-scenariusza-testowego/>

# Typy testów ISTQB

- **Testowanie funkcjonalne**
- **Testowanie нефunkcjonalne sprawdzają:**
  - **wydajność**
  - **kompatybilność**
  - **użyteczność**
  - **niezawodność**
  - **zabezpieczenia**
  - **utrzymywalność**
  - **przenaszalność**
- **Testowanie czarnoskrzynkowe**
- **Testowanie białoskrzynkowe**
- **Testowanie potwierdzające**
- **Testowanie regresji**

# Poziomy testów ISTQB

- **Testowanie modułowe (testy jednostkowe)**
- **Testowanie integracji modułów**
- **Testowanie systemowe**
- **Testowanie integracji systemów**
- **Testowanie akceptacyjne**

# Postman

Postman oferuje programistom platformę API do projektowania, budowania, testowania i współpracy nad interfejsami API

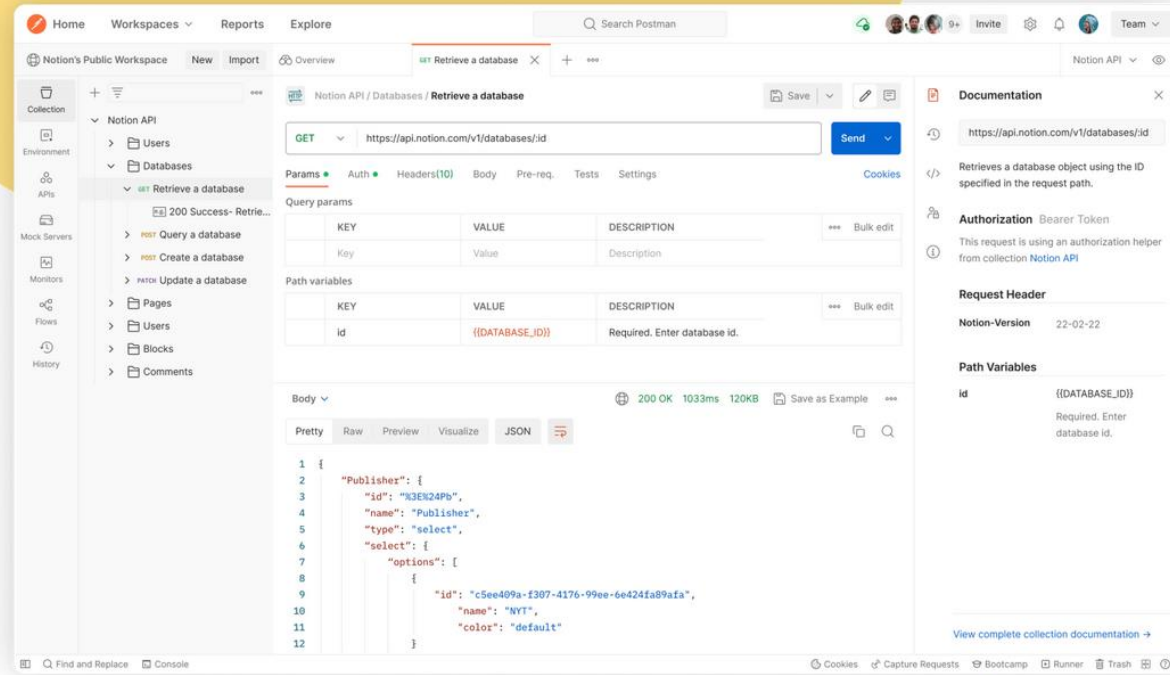


## Build APIs together

Over 30 million developers use Postman. Get started by signing up or downloading the desktop app.

 [Sign Up for Free](#)

Download the desktop app for



<https://www.postman.com/>

# Apache Jmeter

bezpłatne narzędzie do testowania obciążenia i wydajności o otwartym kodzie źródłowym



## About

- [Overview](#)
- [License](#)

## Download

- [Download Releases](#)
- [Release Notes](#)

## Documentation

- [Get Started](#)
- [User Manual](#)
- [Best Practices](#)
- [Component Reference](#)
- [Functions Reference](#)
- [Properties Reference](#)
- [Change History](#)
- [Javadocs](#)
- [JMeter Wiki](#)
- [FAQ \(Wiki\)](#)

## Materials

## Apache JMeter™



The **Apache JMeter™** application is open source software, a 100% pure Java application designed to load test functional behavior and measure performance. It was originally designed for testing Web Applications but has since expanded to other test functions.

## What can I do with it?

Apache JMeter may be used to test performance both on static and dynamic resources, Web dynamic applications. It can be used to simulate a heavy load on a server, group of servers, network or object to test its strength or to analyze overall performance under different load types.

Apache JMeter features include:

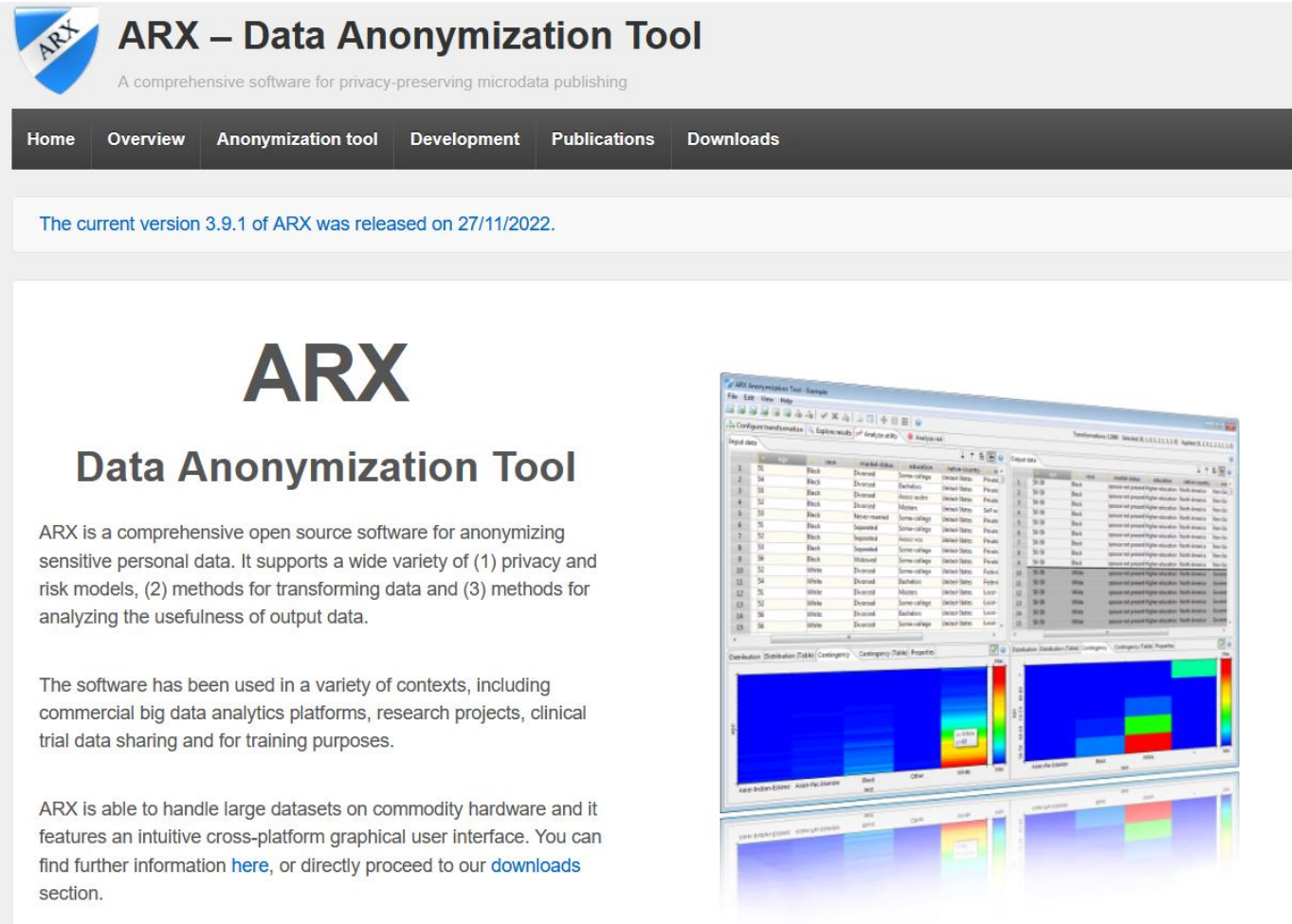
- Ability to load and performance test many different applications/server/protocol types:
  - Web - HTTP, HTTPS (Java, NodeJS, PHP, ASP.NET, ...)
  - SOAP / REST Webservices
  - FTP
  - Database via JDBC
  - LDAP

<https://jmeter.apache.org/index.html>



# ARX

bezpłatne narzędzie do anonimizacji, testuje strony internetowe zawierające wrażliwe dane, np. informacje bankowe



The screenshot shows the ARX website homepage. At the top left is the ARX logo, a blue shield with a white 'A' and 'R' and 'X' in a circle. To its right is the text 'ARX – Data Anonymization Tool' and 'A comprehensive software for privacy-preserving microdata publishing'. Below this is a dark navigation bar with links: Home, Overview, Anonymization tool, Development, Publications, Downloads. A light blue banner below the navigation bar states 'The current version 3.9.1 of ARX was released on 27/11/2022.' The main content area features the ARX logo and the title 'Data Anonymization Tool'. Below this is a paragraph: 'ARX is a comprehensive open source software for anonymizing sensitive personal data. It supports a wide variety of (1) privacy and risk models, (2) methods for transforming data and (3) methods for analyzing the usefulness of output data.' This is followed by another paragraph: 'The software has been used in a variety of contexts, including commercial big data analytics platforms, research projects, clinical trial data sharing and for training purposes.' The final paragraph states: 'ARX is able to handle large datasets on commodity hardware and it features an intuitive cross-platform graphical user interface. You can find further information [here](#), or directly proceed to our [downloads](#) section.' To the right of the text is a screenshot of the ARX software interface. The interface shows a window titled 'ARX Anonymization Tool - Sample'. It has a menu bar (File, Edit, Help) and a toolbar. Below the toolbar are tabs for 'Configure transformation', 'Explore results', 'Analyze utility', and 'Analytics'. The main area is divided into two panes: 'Input data' and 'Output data'. Both panes show a table with columns for 'id', 'sex', 'marital status', 'education', 'income bracket', 'occupation', 'region', and 'country'. The 'Output data' pane shows the same data but with some values replaced by asterisks, indicating anonymization. Below the tables are two heatmaps showing the distribution of data across different categories. The bottom of the interface shows a 'Data Distribution Editor' and 'Add New Statistics' buttons.

<https://arx.deidentifier.org/>

# Jira

narzędzie do śledzenia defektów



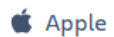
## Connect every team, task, and project together with Jira

Work email

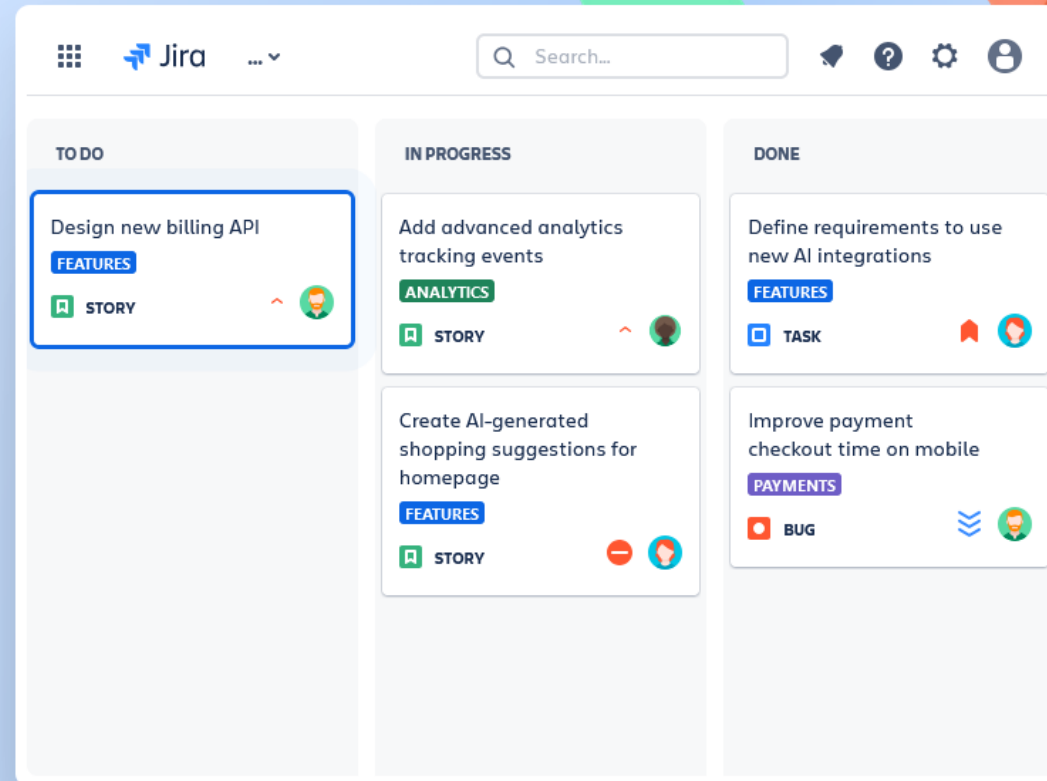
I agree to the [Atlassian Customer Agreement](#), which incorporates by reference the [AI Product-Specific Terms](#), and acknowledge the [Privacy Policy](#).

Get Jira free

Or sign up with



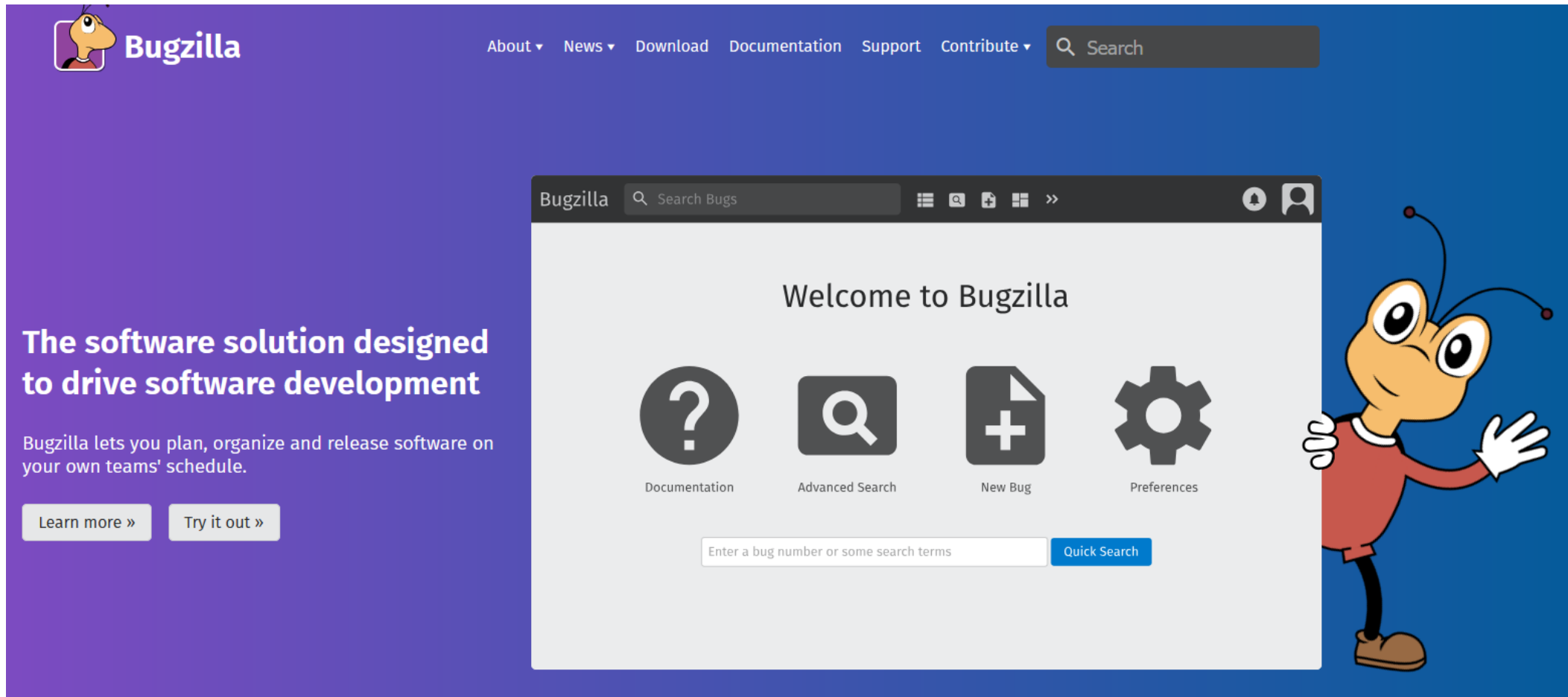
Trying to access Jira? [Log in](#)



<https://www.atlassian.com/pl/software/jira>

# BugZilla

bezpłatne narzędzie do śledzenia defektów



The image shows a screenshot of the Bugzilla website homepage. The background is a gradient of purple and blue. In the top left corner, there is the Bugzilla logo, which is a cartoon bug character, followed by the text "Bugzilla". To the right of the logo is a navigation menu with links: "About", "News", "Download", "Documentation", "Support", and "Contribute". Further right is a search bar with a magnifying glass icon and the text "Search".

Below the navigation menu, on the left side, is a large text block: "The software solution designed to drive software development". Below this text is a smaller line: "Bugzilla lets you plan, organize and release software on your own teams' schedule." At the bottom of this section are two buttons: "Learn more »" and "Try it out »".

In the center of the page is a large white rectangular area representing a browser window. The browser window has a title bar that says "Bugzilla" and a search bar that says "Search Bugs". Below the title bar, the text "Welcome to Bugzilla" is displayed. Underneath this text are four icons in a row: a question mark (Documentation), a magnifying glass (Advanced Search), a plus sign on a document (New Bug), and a gear (Preferences). Below these icons are four corresponding labels: "Documentation", "Advanced Search", "New Bug", and "Preferences". At the bottom of the browser window is a search input field with the placeholder text "Enter a bug number or some search terms" and a blue button labeled "Quick Search".

On the right side of the browser window, a cartoon bug character is peeking out from behind the edge, waving with its right hand. The bug is orange with a red shirt and brown shoes.

<https://www.bugzilla.org/>

# Selenium

bezpłatne narzędzie do automatyzacji przypadków testowych



[About](#) ▾ [Downloads](#) [Documentation](#) [Projects](#) [Support](#) [Blog](#) [English](#) ▾

Registrations open for Appium Conf 2024 online - live on 13 Sep. [Register now!](#)

## Selenium automates browsers. That's it!

What you do with that power is entirely up to you.

Primarily it is for automating web applications for testing purposes, but is certainly not limited to just that. Boring web-based administration tasks can (and should) also be automated as well.

### Getting Started



Selenium WebDriver



Selenium IDE

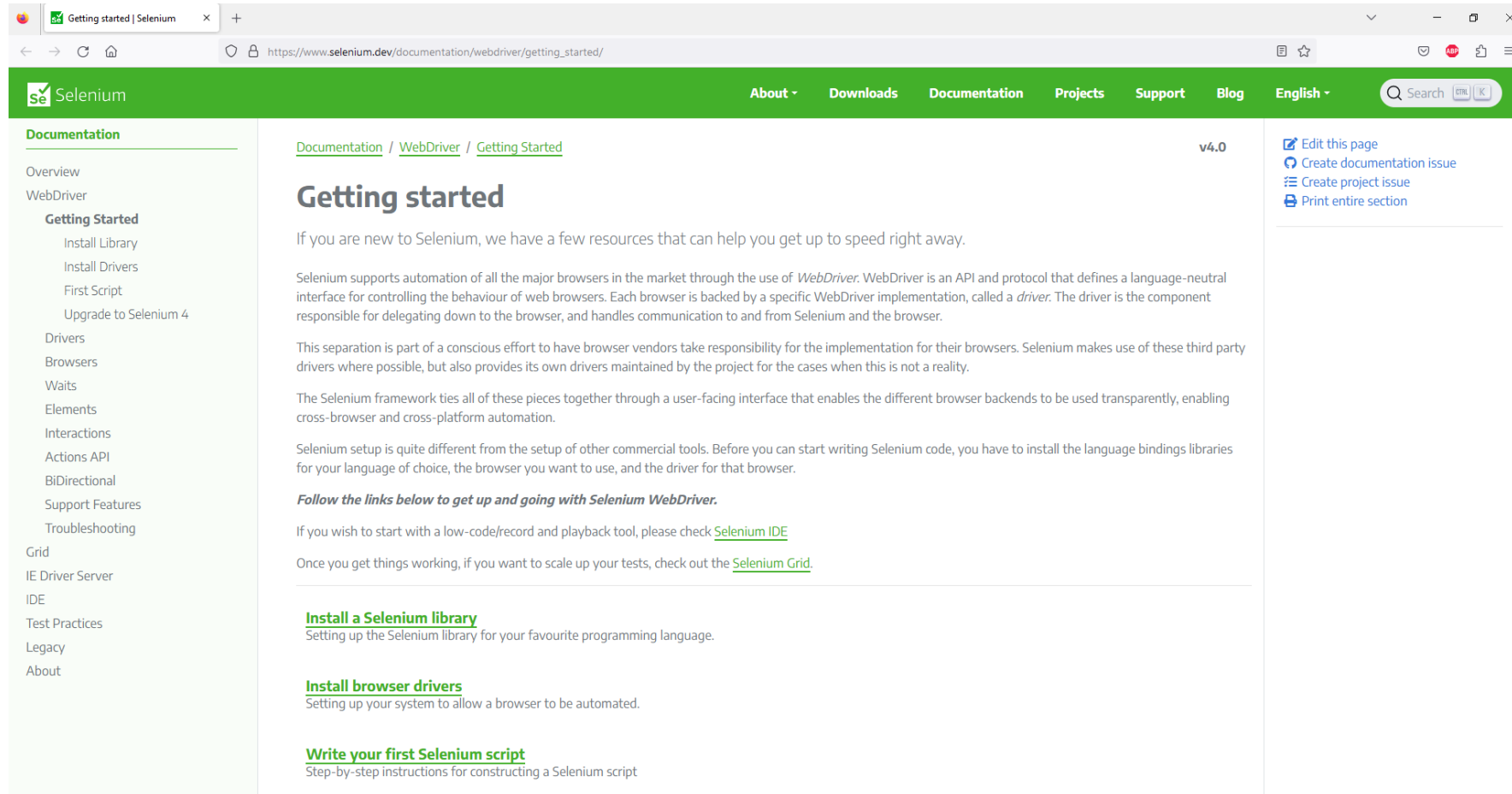


Selenium Grid

<https://www.selenium.dev/>

# testy Selenium

opis instalacji oraz użytkowania Selenium




The screenshot shows a web browser window displaying the Selenium documentation page for 'Getting started'. The browser's address bar shows the URL [https://www.selenium.dev/documentation/webdriver/getting\\_started/](https://www.selenium.dev/documentation/webdriver/getting_started/). The page has a green header with the Selenium logo and navigation links: About, Downloads, Documentation, Projects, Support, Blog, and English. A search bar is also present in the header. On the left side, there is a navigation menu under the heading 'Documentation', listing various topics such as Overview, WebDriver, Getting Started (with sub-items like Install Library, Install Drivers, First Script, Upgrade to Selenium 4), Drivers, Browsers, Waits, Elements, Interactions, Actions API, BiDirectional, Support Features, Troubleshooting, Grid, IE Driver Server, IDE, Test Practices, Legacy, and About. The main content area features the title 'Getting started' and a version number 'v4.0'. The text explains that Selenium supports automation of major browsers through the use of WebDriver, which is an API and protocol that defines a language-neutral interface for controlling the behavior of web browsers. It mentions that each browser is backed by a specific WebDriver implementation, called a driver, which is responsible for delegating down to the browser and handling communication. The page also notes that Selenium makes use of third-party drivers where possible but also provides its own drivers. It further states that Selenium setup is quite different from other commercial tools, requiring the installation of language bindings libraries. A bolded instruction reads: 'Follow the links below to get up and going with Selenium WebDriver.' Below this, it suggests checking Selenium IDE for low-code/record and playback tools, and Selenium Grid for scaling tests. Three sections are highlighted with green underlines: 'Install a Selenium library' (setting up the library for a programming language), 'Install browser drivers' (setting up the system for browser automation), and 'Write your first Selenium script' (step-by-step instructions for constructing a Selenium script). On the right side of the page, there are three utility links: 'Edit this page', 'Create documentation issue', and 'Print entire section'.

[https://www.selenium.dev/documentation/webdriver/getting\\_started/](https://www.selenium.dev/documentation/webdriver/getting_started/)

# Selenium IDE

dodatek do przeglądarek Chrome, Firefox i Edge, który umożliwia proste nagrywanie i odtwarzanie interakcji z przeglądarką

Getting Started



Selenium IDE

If you want to create quick bug reproduction scripts, create scripts to aid in automation-aided exploratory testing, then you want to use Selenium IDE, a Chrome, Firefox and Edge add-on that will do simple record-and-playback of interactions with the browser.

[READ MORE](#)


## Selenium IDE

Open source record and playback test automation for the web

[CHROME DOWNLOAD](#) [FIREFOX DOWNLOAD](#) [LATEST ZIP](#)

☆ Star 2,841

Firefox Browser **ADD-ONS** Extensions Themes More...




Selenium IDE  
by Selenium


⚠ This add-on is not actively monitored for security by Mozilla. Make sure you trust it before installing.  
[Learn more](#)

Selenium IDE is an integrated development environment for Selenium tests. It is implemented as a Firefox extension, and allows you to record, edit, and debug tests.

[Add to Firefox](#)

Rozszerzenia

 Selenium IDE  
Nie może odczytywać ani zmieniać danych na tej witrynie

 React Developer Tools  
Nie może odczytywać ani zmieniać danych na tej witrynie


Zarządzaj rozszerzeniami

Rozszerzenie: (Selenium IDE) - Selenium IDE — Mozilla Firefox

Project: [Icons]

Tests [Icons]

Search tests... [Search Icon] Playback base URL [Dropdown]

 **Welcome to Selenium IDE!**  
Version 3.17.4

What would you like to do?

- Record a new test in a new project**
- Open an existing project
- Create a new project
- Close Selenium IDE

To learn more on Selenium IDE and how to use it visit the [the Selenium IDE project page](#).

# testy Selenium

## testowana strona colors.html




plik colors.html skopiowany jest do folderu htdocs/tests/color/colors.html (Xampp)

Xampp (Apache) jest uruchomiony, strona jest dostępna lokalnie:  
<http://localhost/tests/color/colors.html>






## Kolory

Example

Color	HEX	RGB	Color
	#FF0000	rgb(255,0,0)	Red
	#00FF00	rgb(0,255,0)	Green
	#0000FF	rgb(0,0,255)	Blue

Try it Yourself »

Example

Color	HEX	RGB	Color
	#FF0000	rgb(255,0,0)	Red
	#00FF00	rgb(0,255,0)	Green
	#0000FF	rgb(0,0,255)	Blue

Try it Yourself »

obrazek o nazwie "w3schools\_colors.png"  
<https://www.w3schools.com/colors/default.asp>

kopia powyższego obrazka napisana w html i css

Selenium aby przetestować stronę pobiera ją z lokalnego serwera Apache



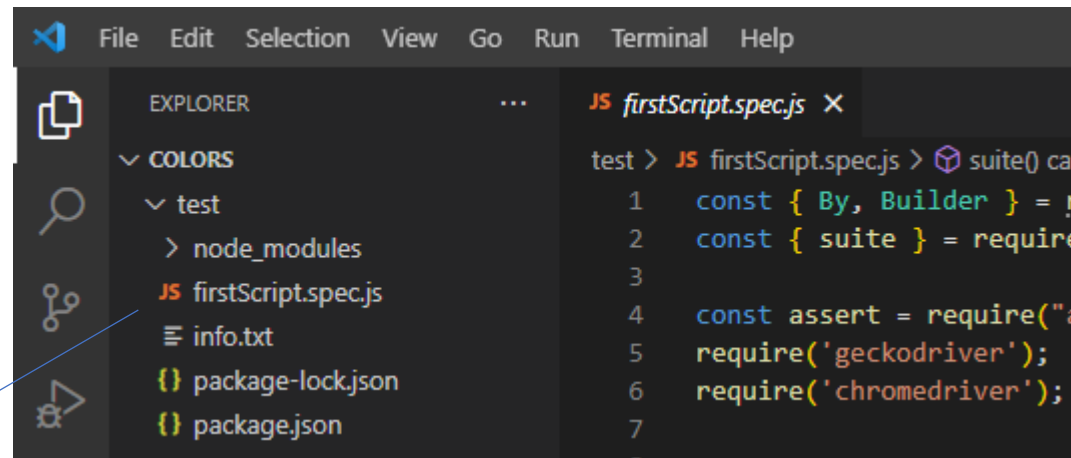
# testy Selenium

// komendy instalacyjne w terminalu Visual Studio Code w folderze colors/test

```
npm install selenium-webdriver  
npm install mocha  
npm install geckodriver --save;  
npm install chromedriver --save;
```

```
// driver selenium  
// driver mocha - puszczanie testu  
// instalacja drivera do firefoxa  
// instalacja drivera do chrome
```

Visual Studio Code



```
File Edit Selection View Go Run Terminal Help  
EXPLORER  
COLORS  
test  
node_modules  
JS firstScript.spec.js  
info.txt  
package-lock.json  
package.json  
... JS firstScript.spec.js X  
test > JS firstScript.spec.js > suite() call  
1 const { By, Builder } = require('selenium-webdriver');  
2 const { suite } = require('mocha');  
3  
4 const assert = require('assert');  
5 require('geckodriver');  
6 require('chromedriver');  
7
```

plik z testami, które trzeba napisać: firstScript.spec.js

# testy Selenium

strona testowana colors.html 1/2

plik z testami firstScript.spec.js

```
JS firstScript.spec.js X
test > JS firstScript.spec.js > ...
1  const { By, Builder } = require('selenium-webdriver');
2  const { suite } = require('selenium-webdriver/testing');
3
4  const assert = require("assert");
5  require('geckodriver');
6  require('chromedriver');
7
8
9  suite(function(env) {
10     describe('First script', function() {
11         let driver;
12
13         before(async function() {
14             driver = await new Builder().forBrowser('chrome').build();
15         });
16
17         after(async () => await driver.quit());
18
19         it('First Selenium script', async function() {
20             await driver.get('http://localhost/tests/color/colors.html');
21
22             let blueXPath = '/html/body/div/table/tbody/tr[4]/td[4]';
23             let blue = await driver.findElement(By.xpath(blueXPath));
24
25             let value = await blue.getText();
26             assert.equal("Blue", value);
27
28             let btn = await driver.findElement(By.id('button'));
29             let btnValue = await btn.getText();
30             assert.equal("Try it Yourself »", btnValue);
31
32             let cont = driver.findElement(By.css("#container"));
33             let contValue = await cont.getCssValue("width");
34             assert.equal("1301px", contValue);
35
36             // imgSrc = 'http://localhost/tests/color/w3schools_colors.png'
37             let imgSrc = await driver.findElement(By.tagName('img')).getAttribute("src");
38             // wyciąga nazwę pliku
39             var filename = imgSrc.replace(/^.*(\\\/)/, '');
40             assert.equal("w3schools_colors.png", filename);
41
42         });
43     });
44 });
45
```

test szerokości kontenera

pobieranie strony do testów

```
colors.html X
colors.html > html > head > style > #button
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <meta charset='utf-8'>
5  <title>colors</title>
6  <style>
7  #container {
8      background-color: #231233235;
9      width: 1301px;
10     height: 325px;
11 }
12 #example{
13     margin:0;
14     padding:20px;
15     font-size: 23px;
16     font-family: Arial, Helvetica, sans-serif;
17 }
18 #table{
19     border: solid 1px;
20     border-color: #c0c0c0;
21     width: 97%;
22     margin-left: auto;
23     margin-right: auto;
24     font-family: arial, sans-serif;
25     border-collapse: collapse;
26 }
27 td, th {
28     text-align: left;
29     padding: 10px;
30     width:100px;
31 }
32 tr:nth-child(odd) {
33     background-color: #ffffff;
34 }
35 #red{
36     background-color: #ff0000;
37 }
38 #green{
39     background-color: #00ff00;
40 }
41 #blue{
42     background-color: #0000ff;
43 }
44 #button{
45     background-color: #515298;
46     border: 0px;
47     color: #white;
48     margin-left: 20px;
49     margin-top: 20px;
50     width:147px;
51     height: 38px;
52     font-size: 16px;
53     border-radius: 8px;
54 }
55 </style>
56 </head>
```

# testy Selenium

strona testowana colors.html 2/2

plik z testami firstScript.spec.js

```
JS firstScript.spec.js X
test > JS firstScript.spec.js > ...
1  const { By, Builder } = require('selenium-webdriver');
2  const { suite } = require('selenium-webdriver/testing');
3
4  const assert = require("assert");
5  require('geckodriver');
6  require('chromedriver');
7
8
9  suite(function(env) {
10   describe('First script', function() {
11     let driver;
12
13     before(async function() {
14       driver = await new Builder().forBrowser('chrome').build();
15     });
16
17     after(async () => await driver.quit());
18
19     it('First Selenium script', async function() {
20       await driver.get('http://localhost/tests/color/colors.html');
21
22       let blueXPath = '/html/body/div/table/tbody/tr[4]/td[4]';
23       let blue = await driver.findElement(By.xpath(blueXPath));
24
25       let value = await blue.getText();
26       assert.equal("Blue", value);
27
28       let btn = await driver.findElement(By.id('button'));
29       let btnValue = await btn.getText();
30       assert.equal("Try it Yourself »", btnValue);
31
32       let cont = driver.findElement(By.css("#container"));
33       let contValue = await cont.getCssValue("width");
34       assert.equal("1301px", contValue);
35
36       // imgSrc = 'http://localhost/tests/color/w3schools_colors.png'
37       let imgSrc = await driver.findElement(By.tagName('img')).getAttribute("src");
38       // wyciąga nazwę pliku
39       var filename = imgSrc.replace(/^.*[\\\/], '');
40       assert.equal("w3schools_colors.png", filename);
41
42     });
43
44   });
45 });
```

test nazwy  
wyświetlanego na  
stronie pliku

```
57 <body>
58 <h1>Kolory</h1>
59 
60 <div id="container">
61   <p id="example">Example</p>
62   <table id="table">
63     <thead>
64       <tr>
65         <th>Color</th>
66         <th>HEX</th>
67         <th>RGB</th>
68         <th>Color</th>
69       </tr>
70     <tbody>
71       <tr>
72         <td id="red"></td>
73         <td>#FF0000</td>
74         <td>rgb(255,0,0)</td>
75         <td>Red</td>
76       </tr>
77       <tr>
78         <td id="green"></td>
79         <td>#00FF00</td>
80         <td>rgb(0,255,0)</td>
81         <td>Green</td>
82       </tr>
83       <tr>
84         <td id="blue"></td>
85         <td>#0000FF</td>
86         <td>rgb(0,0,255)</td>
87         <td>Blue</td>
88       </tr>
89     </tbody>
90   </table>
91   <button id="button">Try it Yourself&nbsp;&#187;</button>
92 </div>
93 </body>
94 </html>
```




# testy Selenium

plik z testami firstScript.spec.js

```
JS firstScript.spec.js X
test > JS firstScript.spec.js > ...
1  const { By, Builder } = require('selenium-webdriver');
2  const { suite } = require('selenium-webdriver/testing');
3
4  const assert = require("assert");
5  require('geckodriver');
6  require('chromedriver');
7
8
9  suite(function(env) {
10   describe('First script', function() {
11     let driver;
12
13     before(async function() {
14       driver = await new Builder().forBrowser('chrome').build();
15     });
16
17     after(async () => await driver.quit());
18
19     it('First Selenium script', async function() {
20       await driver.get('http://localhost/tests/color/colors.html');
21
22       let blueXpath = '/html/body/div/table/tbody/tr[4]/td[4]';
23       let blue = await driver.findElement(By.xpath(blueXpath));
24
25       let value = await blue.getText();
26       assert.equal("Blue", value);
27
28       let btn = await driver.findElement(By.id('button'));
29       let btnValue = await btn.getText();
30       assert.equal("Try it Yourself »", btnValue);
31
32       let cont = driver.findElement(By.css("#container"));
33       let contValue = await cont.getCssValue("width");
34       assert.equal("1301px", contValue);
35
36       // imgSrc = 'http://localhost/tests/color/w3schools_colors.png'
37       let imgSrc = await driver.findElement(By.tagName('img')).getAttribute("src");
38       // wyciąga nazwę pliku
39       var filename = imgSrc.replace(/^.*[\\\/]/, '');
40       assert.equal("w3schools_colors.png", filename);
41     });
42   });
43
44 });
45 });
```


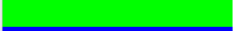

## Kolory

Example

Color	HEX	RGB	Color
	#FF0000	rgb(255,0,0)	Red
	#00FF00	rgb(0,255,0)	Green
	#0000FF	rgb(0,0,255)	Blue

Try it Yourself »

Example

Color	HEX	RGB	Color
	#FF0000	rgb(255,0,0)	Red
	#00FF00	rgb(0,255,0)	Green
	#0000FF	rgb(0,0,255)	Blue

Try it Yourself »

test treści jednej z komórek tabeli

test napisu na buttonie

# testy Selenium

polecenie uruchamiające test  
(w folderze test)

```
PS D:\colors\test> npx mocha .\firstScript.spec.js
```

```
firstScript.spec.js - colors - Visual Studio Code

EXPLORER
  COLORS
  test
    node_modules
    JS firstScript.spec.js
    info.txt
    package-lock.json
    package.json

JS firstScript.spec.js
test > JS firstScript.spec.js > suite() callback > describe('First script') callback > it('First Selenium script') callback
1  const { By, Builder } = require('selenium-webdriver');
2  const { suite } = require('..selenium-webdriver/testing');
3
4  const assert = require("assert");
5  require('geckodriver');
6  require('chromedriver');
7

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

PS D:\colors> cd .\test\
PS D:\colors\test> npx mocha .\firstScript.spec.js
[INFO] Searching for WebDriver executables installed on the current system...
[INFO] ... located chrome
[INFO] ... located firefox
[INFO] Running tests against [chrome, firefox]

[chrome]
  First script

DevTools listening on ws://127.0.0.1:59949/devtools/browser/aea6f5f9-53bc-49b9-b020-fbc3a3b774c7
  ✓ First Selenium script (393ms)

[firefox]
  First script

DevTools listening on ws://127.0.0.1:59965/devtools/browser/c0b1f928-f8d8-4962-9c2e-a23b66b49913
  ✓ First Selenium script (400ms)

2 passing (4s)

PS D:\colors\test>
```

test uruchomiony w  
przeglądarce Chrom

test uruchomiony w  
przeglądarce Firefox

# Playwright

**Playwright** enables reliable end-to-end testing for modern web apps.

GET STARTED

Star 66k+



**Any browser • Any platform • One API**

**Cross-browser.** Playwright supports all modern rendering engines including Chromium, WebKit, and Firefox.

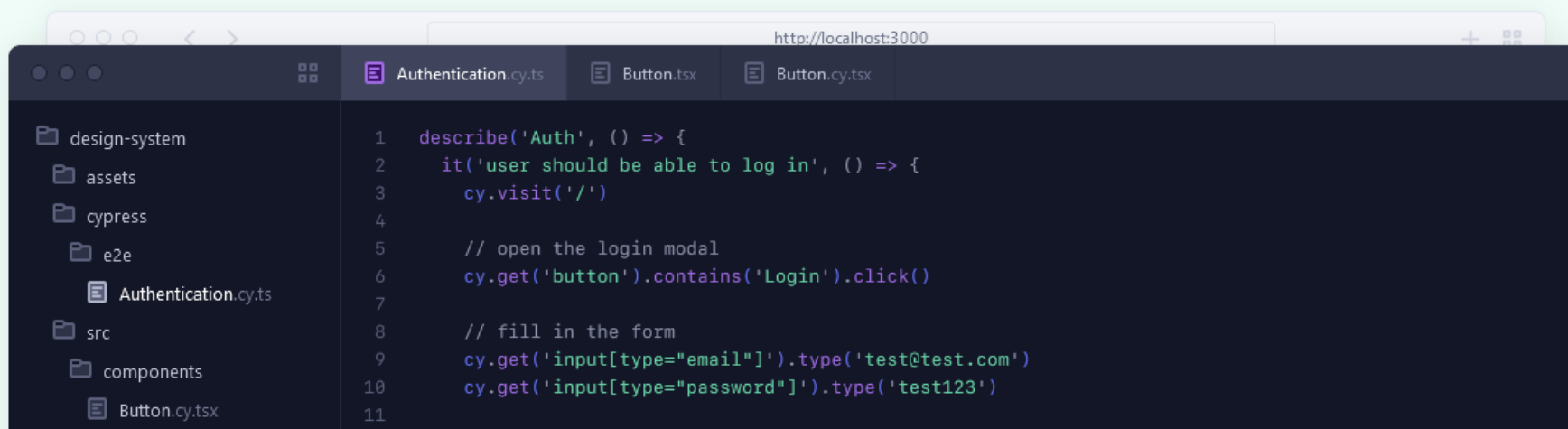
<https://playwright.dev/>

# Cypress

[Product](#)[Docs](#)[Community](#)[Company](#)[Pricing](#)[Contact sales >](#)[Log in >](#)[Sign up >](#)

## Test. Automate. Accelerate.

With Cypress, you can easily create tests for your modern web applications, debug them visually, and automatically run them in your continuous integration builds.

[> npm install cypress](#)[Compare plans](#)

<https://www.cypress.io/>



# PMD

bezpłatne narzędzie do analizy statycznej kodu


🏠 PMD Source Code Analyzer


[Home](#) [Documentation](#) [Bugs](#) [About](#) [News](#) [Downloads](#) [Support](#) [Plugins](#)



## PMD

An extensible cross-language static code analyzer.

 Download

 Documentation

Latest Version: 7.5.0 (30-August-2024)

[Release Notes](#) | [Source](#)

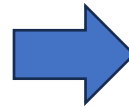
<https://pmd.github.io/>

# Postman – darmowe API

wykaz publicznych, darmowych API

## Try Public APIs for free

The Public APIs repository is manually curated by community members like you and folks working at [APILayer](#). It includes an extensive list of public APIs from many domains that you can use for your own products. Consider it a treasure trove of APIs well-managed by the community over the years.



przykładowy adres API, za pomocą którego można uzyskać informacje o książce „The lord of the rings”

Open Library Search API

The screenshot shows the Open Library Search API page. At the top, there's a navigation bar with "OPEN LIBRARY" and "Moje książki". Below that, there's a search bar and a "Zaloguj się" button. The main content area has the title "Open Library Search API" and a list of features. The first feature is circled in red in the original image.

INTERNET ARCHIVE Dorzuć się

OPEN LIBRARY Moje książki Przeglądaj Wszystkie Szukaj Zaloguj się Załóż konto

### Open Library Search API

Last edited by [Drini](#) 10 sierpnia 2024 | [Historia](#) [Edytuj](#)

The Open Library Search API is one of the most convenient and complete ways to retrieve book data on Open Library. The API:

1. Is able to return data for **multiple** books in a single request/response
2. Returns both Work level information about the book (like author info, first publish year, etc), as well as Edition level information (like title, identifiers, covers, etc)
3. Author IDs are returned which you can use to fetch the author's image, if available
4. Options are available to return Book Availability along with the response.
5. Powerful sorting options are available, such as star ratings, publication date, and number of editions.

### Examples

The URL format for API is simple. Take the search URL and add `.json` to the end. Eg:

- <https://openlibrary.org/search.json?q=the+lord+of+the+rings>
- <https://openlibrary.org/search.json?title=the+lord+of+the+rings>
- <https://openlibrary.org/search.json?author=tolkien&sort=new>
- <https://openlibrary.org/search.json?q=the+lord+of+the+rings&page=2>
- <https://openlibrary.org/search/authors.json?q=twain>

# Postman – połączenie z publicznym API

The screenshot shows the Postman interface with a GET request to `https://openlibrary.org/search.json?q=the+lord+of+the+rings`. The response is a JSON object with the following structure:

```
1 {
2   "numFound": 747,
3   "start": 0,
4   "numFoundExact": true,
5   "docs": [
6     {
7       "author_alternative_name": [
8         "John Ronald Reuel Tolkien",
9         "John R. R. Tolkien",
10        "J. R. R. Tolkien"
11      ]
12    }
13  ]
14 }
```

Annotations in the image include a red circle around the URL in the request bar, a red circle around the JSON response body, and an orange arrow pointing from the text 'komenda GET z adresem API' to the request bar.

komenda GET z adresem API

zwrócona informacja w postaci pliku w formacie json

## test jednostkowy – unit test

**Test jednostkowy** (ang. *unit test*) – metoda testowania tworzonego oprogramowania poprzez wykonywanie testów weryfikujących poprawność działania pojedynczych elementów (jednostek) programu – np. metod lub obiektów w programowaniu obiektowym lub procedur w programowaniu proceduralnym. Testowany fragment programu poddawany jest testowi, który wykonuje go i porównuje wynik (np. zwrócone wartości, stan obiektu, zgłoszone wyjątki) z oczekiwanymi wynikami – tak pozytywnymi, jak i negatywnymi (niepowodzenie działania kodu w określonych sytuacjach również może podlegać testowaniu).

# test jednostkowy – unit test w Visual Studio

Learn Discover Product documentation Development languages Topics

Visual Studio IDE Edit Build Debug Test Deploy Common tasks Troubleshooting Resources

Version  
Visual Studio 2022  
Filter by title

Run a unit test as a 64-bit process  
Configure unit tests by using a .runsettings file  
Test Execution with Hot Reload  
> Write unit tests for managed code  
Unit tests for C/C++ code  
Write unit tests for C/C++ code  
Use the Microsoft Unit Testing Framework for C++  
Use Google C++ Testing Framework  
Use Boost.Test  
Use CTest  
Write unit tests for C/C++ DLLs  
Walkthrough: Writing Unit tests for C++ DLLs

Learn / Visual Studio / Testing /

## Write unit tests for C/C++ in Visual Studio

Article • 06/05/2024 • 16 contributors Feedback

**In this article**

- Basic test workflow
- Use CodeLens
- Related content

You can write and run your C++ unit tests by using the **Test Explorer** window. It works just like it does for other languages. For more information about using **Test Explorer**, see [Run unit tests with Test Explorer](#).

**Note**  
Some features such as Live Unit Testing, Coded UI Tests and IntelliTest aren't supported for C++.

<https://learn.microsoft.com/en-us/visualstudio/test/writing-unit-tests-for-cpp?view=vs-2022>

Learn / Visual Studio / Testing /

## Microsoft.VisualStudio.TestTools.CppUnitTestFramework API reference

Article • 03/10/2023 • 15 contributors Feedback

### In this article

- In this topic
- CppUnitTest.h
- CppUnitTestAssert.h
- CppUnitTestLogger.h
- Show 2 more

This topic lists the public members of the `Microsoft::VisualStudio::CppUnitTestFramework` namespace. Use these APIs to write C++ unit tests based on the Microsoft Native Unit Test Framework. There is a [Usage Example](#) at the end of the topic.

<https://learn.microsoft.com/en-us/visualstudio/test/microsoft-visualstudio-testtools-cppunittestframework-api-reference?view=vs-2022>

# test jednostkowy – unit test w Visual Studio

projekt konsolowy C++  
*framework\_testing*,  
który będzie testowany

```
framework_testing.cpp  + x
framework_testing  -  (Globalny zasięg)
1
2  #include <iostream>
3
4  using namespace std;
5
6  class MyClass
7  {
8  public:
9      string name;
10     MyClass(string n) {
11         name = n;
12     };
13     string getName() {
14         return name;
15     };
16 };
17
18 int main()
19 {
20     MyClass myClass("Marek");
21     cout << myClass.getName();
22 }
```

```
Konsola debugowania programu Microsoft Visual Studio
Marek
D:\arch\TechnikProgramista\testowanie i dokumentowanie
\Debug\framework_testing.exe (proces 13852) zakonczono
Nacisnij dowolny klawisz, aby zamknac to okno...
```

Eksplorator rozwiązań

Przeszukaj: Eksplorator rozwiązań (Ctrl+;)

Rozwiązanie „framework\_testing” (2 z 2 projektów)

- framework\_testing
  - Odwołania
  - Zależności zewnętrzne
  - Pliki nagłówkowe
  - Pliki zasobów
  - Pliki źródłowe

# Unit Testing Framework for C++

**Dodawanie nowego projektu**

Ostatnie szablony projektów

- Projekt natywnego testu jednostkowego C++
- Aplikacja konsoli C++
- Aplikacja WPF C#
- Aplikacja WPF (.NET Framework) C#
- Pusty projekt C++
- Aplikacja mobilna (Xamarin.Forms) C#

test

C++ Wszystkie platformy Wszystkie typy projektów

Google Test  
Pisz testy jednostkowe języka C++ za pomocą platformy Google Test. Obejmuje kopie biblioteki Google Test, która ma być używana.  
C++ Windows Test

**Projekt natywnego testu jednostkowego**  
Zapisz testy jednostkowe języka C++ przy użyciu natywnej platformy CppUnitTest firmy Microsoft.  
C++ Windows Test

Projekt elementów udostępnionych  
Projekt Elementy udostępnione jest używany do udostępniania plików między wieloma projektami.  
C++ Windows Android OS Linux Klasyczny Konsola Biblioteka UWP Gry Mobilny

Inne wyniki oparte na wyszukiwaniu

- MSTest Playwright Test Project  
A project that contains Playwright tests using MSTest framework, that can run on .NET on Windows, Linux and MacOS.  
C# MSTest Playwright Test
- NUnit Playwright Test Project

Dalej

**Eksplorator rozwiązań**

Przeszukaj: Eksplorator rozwiązań (Ctrl+;)

Rozwiązanie „framework\_testing” (2 z 2 projektów)

- framework\_testing
  - Odwołania
  - Zależności zewnętrzne
  - Pliki nagłówkowe
  - Pliki zasobów
  - Pliki źródłowe
  - framework\_testing\_UnitTest1
    - Odwołania
    - framework\_testing
    - Zależności zewnętrzne
    - Pliki nagłówkowe
      - pch.h
    - Pliki zasobów
    - Pliki źródłowe
      - framework\_testing\_UnitTest1.cpp
      - pch.cpp

prawy przycisk myszy

dodajemy nowy projekt testowy  
*framework\_testing\_UnitTest1*



# Unit Testing Framework for C++

The image illustrates the steps to add a dependency to a project in Visual Studio:

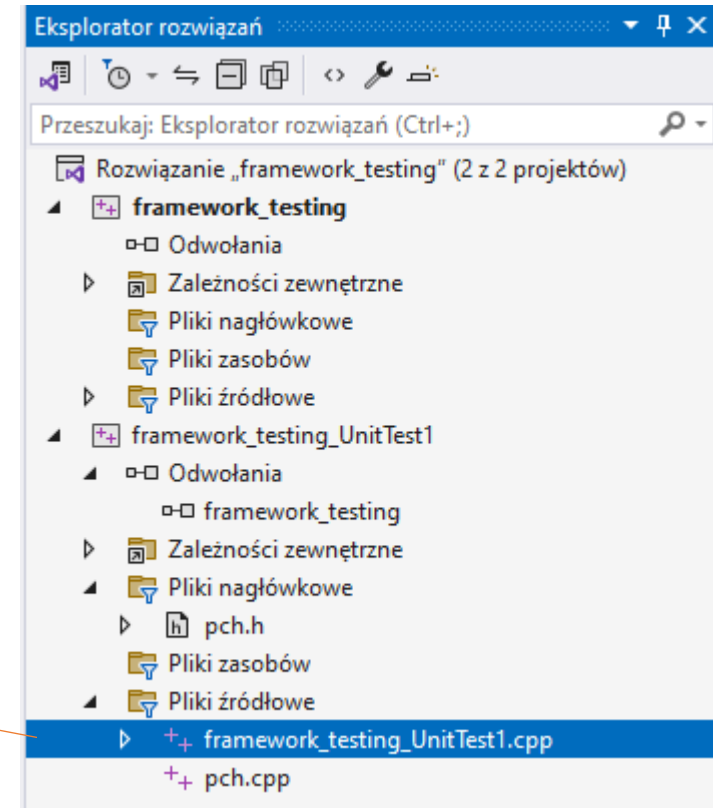
- Dodaj odwołanie dialog:** Shows the 'Projekty' list with 'framework\_testing' selected. The 'Nazwa' field contains 'framework\_testing'. The 'OK' button is highlighted.
- Eksplorator rozwiązań:** Shows the project structure. The 'framework\_testing' dependency is highlighted under the 'framework\_testing\_UnitTest1' project.
- Context menu:** Shows the 'Dodaj' > 'Odwołanie...' option selected.

dodajemy odwołanie do projektu *framework\_testing*

prawy przycisk myszy

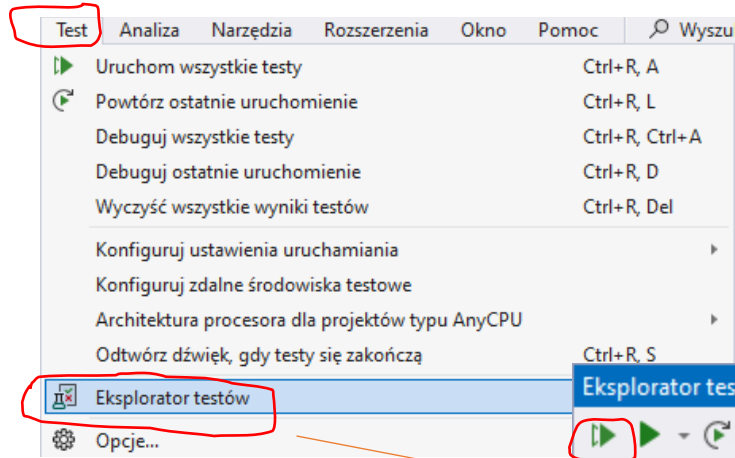
# Unit Testing Framework for C++

```
1 #include "pch.h"
2 #include "CppUnitTest.h"
3 #include "../framework_testing/framework_testing.cpp"
4
5 using namespace Microsoft::VisualStudio::CppUnitTestFramework;
6
7 namespace frameworktestingUnitTest1
8 {
9     TEST_CLASS(frameworktestingUnitTest1)
10    {
11    public:
12
13        TEST_METHOD(TestMethod1)
14        {
15            string name = "Bill";
16            MyClass myClass(name);
17            Assert::AreEqual(expected: name, actual: myClass.getName());
18        }
19    };
20 }
```



piszemy test metody getName()

# Unit Testing Framework for C++



uruchamiamy testy

Zakończono przebieg testu: testy uruchomione w 138 ms: 1 (zakończone pomyślnie: 1, zakończone niepowodzeniem: 0, pominięte: 0) Ostrzeżenia: 0 Błędy: 0

Test	Czas trwa...	Cechy
framework_testing_UnitTest1 (1)	1 ms	
frameworktestingUnitTest1 (1)	1 ms	
frameworktestingUnitTest1 (1)	1 ms	
TestMethod1	1 ms	

Podsumowanie szczegółów testu

- TestMethod1
  - Źródło: [framework\\_testing\\_UnitTest1.cpp](#) wiersz 13
  - Czas trwania: 1 ms

# INF.04 wypożyczalnia filmów

## Część I. Aplikacja konsolowa

NF.04-03-23.06-SG

Korzystając z mechanizmów programowania obiektowego zaprojektuj część logiki systemu wirtualnej wypożyczalni filmów. Zaimplementuj aplikację zawierającą klasę o nazwie *film* z metodami *get*, *set* oraz *inkrementuj*.

Założenia klasy:

- Obiektowy język programowania zgodny z zainstalowanym na stanowisku egzaminacyjnym, jeden z: C++, C#, Java, Python.
- Klasa *film* zawiera dwa pola zawierające tytuł oraz liczbę wypożyczeń. Dla uproszczenia można przyjąć, że tytuł może mieć maksymalnie 20 znaków. Dostęp do obu pól ma jedynie klasa. W przypadku późniejszego rozszerzenia klasy, klasy potomne powinny również mieć dostęp do tych pól.
- Klasa *film* zawiera funkcjonalność:
  - inicjalizację pól wartością pustą lub 0 w zależności od typu
  - metodę ustawiającą tytuł (*ang. setter*)
  - metodę pobierającą tytuł (*ang. getter*)
  - metodę pobierającą liczbę wypożyczeń (*ang. getter*)
  - bezargumentową metodę realizującą inkrementację pola przechowującego liczbę wypożyczeń. Metoda nie zwraca żadnej wartości

Założenia aplikacji:



- Inicjalizacja obiektu i wyświetlenie zawartości jego pól
- Test działania metody *set*, poprzez ustawienie dowolnej wartości dla pola obiektu,
- Test działania metody *get*, poprzez pobranie wartości z pola obiektu i wyświetlenie jej na ekranie
- Test metody inkrementacji pola przechowującego liczbę wypożyczeń zrealizowany w ten sposób, że wyświetlona jest wartość pola przed i po zastosowaniu metody
- Aplikacja powinna być zapisana czytelnie, z zachowaniem zasad czystego formatowania kodu, należy stosować znaczące nazwy pól i metod
- Dokumentacja do aplikacji wykonana zgodnie z wytycznymi z części III zadania egzaminacyjnego.

Kod aplikacji przygotuj do nagrania na płytę. W podfolderze konsola powinno znaleźć się archiwum całego projektu o nazwie *konsola.zip*, plik z kodem źródłowym programu oraz plik wykonywalny, jeżeli istnieje

# INF.04 wypożyczalnia filmów

```
/*
*****
nazwa klasy : Film
pola : tytuł - tytuł filmu
liczbaWypozytczen - liczba wypożyczeń filmu
metody : setTytuł(string tytułFilmu), nic nie zwraca - metoda ustawia pole tytuł
         getTytuł(), tytuł - metoda zwraca pole tytuł
         getLiczbaWypozytczen() , liczbaWypozytczen - metoda zwraca pole liczbaWypozytczen
         inkrementujLiczbeWypozytczen(), nic nie zwraca - metoda inkrementuje pole liczbaWypozytczen
informacje : klasa reprezentuje wypożyczalnię filmu
autor : 1234567890
*****
*/
class Film {
protected:
    string tytuł = "";
    int liczbaWypozytczen = 0;
public:
    void setTytuł(string tytułFilmu) {
        tytuł = tytułFilmu;
    }
    string getTytuł() {
        return tytuł;
    }
    int getLiczbaWypozytczen() {
        return liczbaWypozytczen;
    }
    void inkrementujLiczbeWypozytczen() {
        liczbaWypozytczen++;
    }
};
```

# INF.04 wypożyczalnia filmów – unit testy

The image shows a Visual Studio IDE with a C++ project named 'film'. The main window displays the code for 'filmUnitTest.cpp'. The code defines a test class 'filmUnitTest' with two test methods: 'TestTytułFilmu' and 'TestInkrementacja'. The code is as follows:

```
9 TEST_CLASS(filmUnitTest)
10 {
11     public:
12
13     TEST_METHOD(TestTytułFilmu) { ... }
14
15
16
17
18
19
20
21
22     TEST_METHOD(TestInkrementacja) { ... }
23
24
25
26
27
28
29
30
31 };
32
33
```

An orange arrow points to the test method definitions with the text: "napisz testy jednostkowe w Visual Studio Unit Testing Framework for C++".

The 'Eksplorator testów' (Test Explorer) window is open, showing the results of the test run. The test run was completed in 101 ms with 2 tests passed, 0 failed, and 0 ignored. The test results are as follows:

Test	Czas trwania...	Cechy	Komu...
filmUnitTest (2)	7 ms		
filmUnitTest (2)	7 ms		
TestInkrementacja	7 ms		
TestTytułFilmu	< 1 ms		

The summary for the test group 'filmUnitTest' shows 2 tests in the group and a total execution time of 7 ms. The results are 2 successes and 0 failures.

# INF.04 pesel– unit testy

INF.04-01-24.01-SG

## Część I. Aplikacja konsolowa

Za pomocą narzędzi do tworzenia aplikacji konsolowych zaimplementuj program sprawdzający poprawność numeru PESEL. Program powinien sprawdzać płeć i sumę kontrolną według opisu:

Numer PESEL jest to 11-cyfrowy identyfikator numeryczny.

### Płeć

Informacja o płci osoby zawarta jest na 10. (przedostatniej) pozycji numeru PESEL.

- cyfry 0, 2, 4, 6, 8 (parzyste) – oznaczają płeć żeńską
- cyfry 1, 3, 5, 7, 9 (nieparzyste) – oznaczają płeć męską

### Cyfra kontrolna i sprawdzanie poprawności numeru

Jedenastą cyfrą numeru PESEL jest cyfra kontrolna umożliwiająca kontrolę poprawności identyfikatora. Jest ona wynikiem działania na pierwszych dziesięciu cyfrach.

Algorytm obliczania cyfry kontrolnej na podstawie kolejnych cyfr numeru:

1. Dla kolejnych 10 cyfr numeru PESEL oblicz iloczyn każdej cyfry i jej wagi na podstawie tabeli:

Pozycja cyfry od lewej	1	2	3	4	5	6	7	8	9	10
Waga cyfry	1	3	7	9	1	3	7	9	1	3

Oznacza to, że pierwszą cyfrę numeru PESEL należy pomnożyć przez 1, drugą cyfrę przez 3, trzecią przez 7 itd.

2. Wszystkie iloczyny zsumuj ze sobą i zapisz w zmiennej S
3. Wykonaj operację modulo 10 na sumie S i zapisz w zmiennej M
4. Gdy wartość zmiennej M jest równa 0, to zmiennej R przypisz wartość 0. W przeciwnym przypadku zmiennej R przypisz wartość różnicy 10 i M ( $R=10-M$ )
5. Zmienna R stanowi sumę kontrolną numeru PESEL i musi być równa jedenastej cyfrze numeru PESEL



# INF.04 pesel– unit testy

```
#include "pch.h"  
#include "CppUnitTest.h"  
#include "../pesel_konsola/pesel.cpp"
```

```
using namespace Microsoft::VisualStudio::CppUnitTestFramework;
```

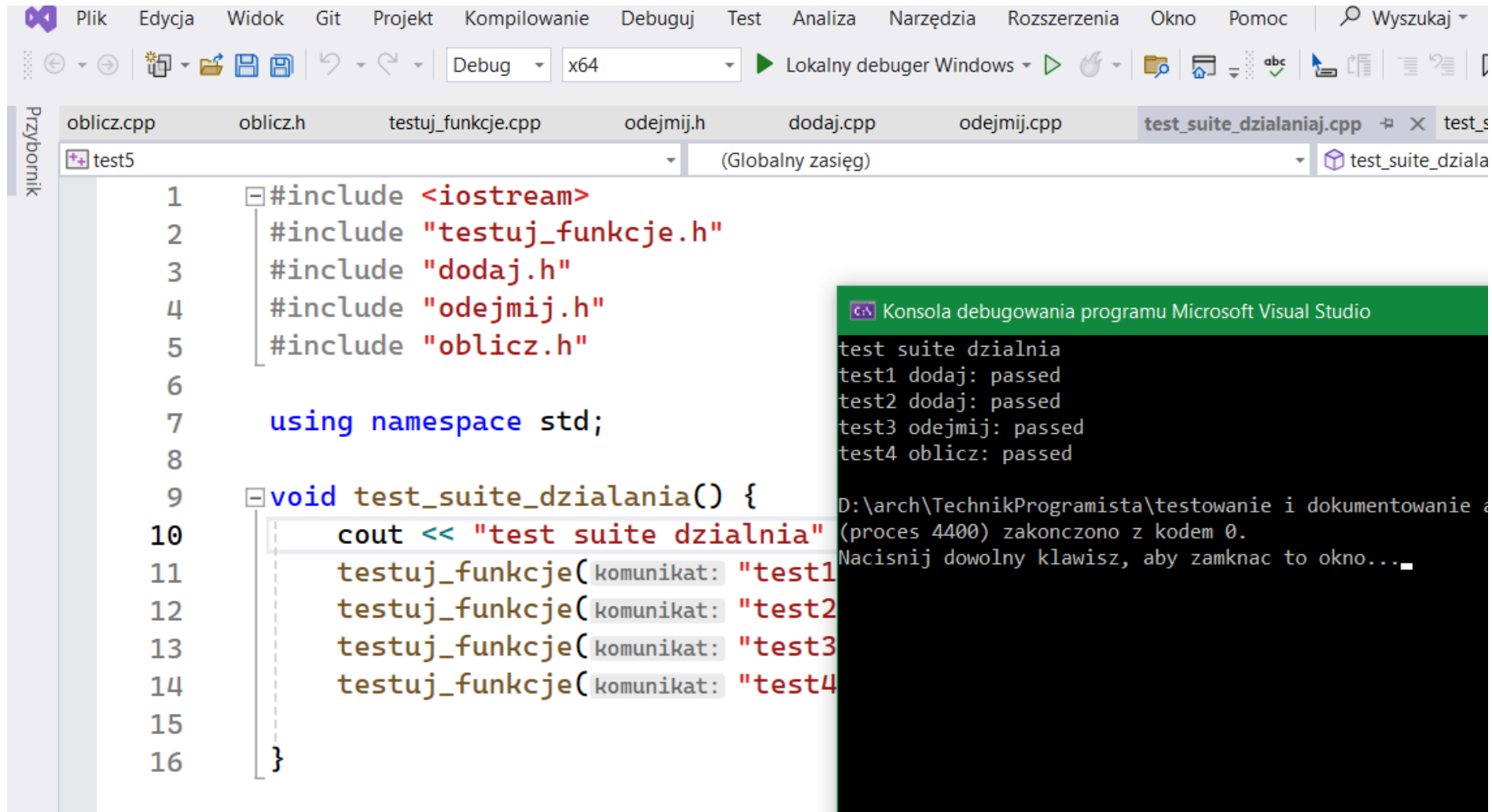
```
namespace peselUnitTest1  
{  
    TEST_CLASS(peselUnitTest1)  
    {  
    public:  
        ✓ TEST_METHOD(getPesel) { ... }  
        ✓ TEST_METHOD(checkGender) { ... }  
        ✓ TEST_METHOD(checkChecksum) { ... }  
    };  
}
```

napisz testy  
jednostkowe w  
Visual Studio Unit  
Testing Framework  
for C++



# test jednostkowy – unit test

własny framework do testowania funkcji



The image shows a screenshot of the Microsoft Visual Studio IDE. The main window displays a C++ source file named `test5` with the following code:

```
1  #include <iostream>
2  #include "testuj_funkcje.h"
3  #include "dodaj.h"
4  #include "odejmij.h"
5  #include "oblicz.h"
6
7  using namespace std;
8
9  void test_suite_dzialania() {
10     cout << "test suite dzialnia"
11     testuj_funkcje(komunikat: "test1
12     testuj_funkcje(komunikat: "test2
13     testuj_funkcje(komunikat: "test3
14     testuj_funkcje(komunikat: "test4
15
16 }
```

Overlaid on the bottom right of the code editor is a window titled "Konsola debugowania programu Microsoft Visual Studio". It displays the output of the test suite:

```
test suite dzialnia
test1 dodaj: passed
test2 dodaj: passed
test3 odejmij: passed
test4 oblicz: passed
```

Below the test results, the console shows the program's exit message:

```
D:\arch\TechnikProgramista\testowanie i dokumentowanie a
(proces 4400) zakonczono z kodem 0.
Nacisnij dowolny klawisz, aby zamknac to okno...
```

# Angular

## Set up testing

The Angular CLI downloads and installs everything you need to test an Angular application with [Jasmine testing framework](#).

The project you create with the CLI is immediately ready to test. Just run the `ng test` CLI command:

```
$ ng test
```

The `ng test` command builds the application in *watch mode*, and launches the [Karma test runner](#).

The console output looks like below:

```
$ 02 11 2022 09:08:28.605:INFO [karma-server]: Karma v6.4.1 server started at
$ 02 11 2022 09:08:28.607:INFO [launcher]: Launching browsers Chrome with concurre
$ 02 11 2022 09:08:28.620:INFO [launcher]: Starting browser Chrome
$ 02 11 2022 09:08:31.312:INFO [Chrome]: Connected on socket -LaEYvD2R7MdcS0-AAAB
$ Chrome: Executed 3 of 3 SUCCESS (0.193 secs / 0.172 secs)
$ TOTAL: 3 SUCCESS
```

<https://angular.dev/guide/testing>

# JavaScript Jest

## JavaScript Testing Framework

JEST 29.7 ▼

[Docs](#) [API](#) [Help](#)

Introduction ▼

Getting Started

Using Matchers

Testing Asynchronous Code

Setup and Teardown

Mock Functions

Jest Platform

Jest Community

More Resources

Guides >

Framework Guides >

Upgrade Guides >

🏠 > Introduction > Getting Started

Version: 29.7

## Getting Started

Install Jest using your favorite package manager:

**npm** Yarn pnpm

```
npm install --save-dev jest
```

Let's get started by writing a test for a hypothetical function that adds two numbers. First, create a `sum.js` file:

```
function sum(a, b) {  
  return a + b;  
}  
module.exports = sum;
```

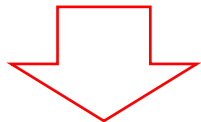
<https://jestjs.io/docs/getting-started>

# Jest – test kalkulatora

```
// utwórz katalog calc w Visual Studio Code  
// przejdź do katalogu calc:  
cd calc  
// zainstaluj jest:  
npm install --save-dev jest  
// zainicjuj projekt w node.js:  
npm init -y
```

```
// zamień właściwość „scripts”  
// w pliku package.json:
```

```
"scripts": {  
  "test": "echo \"Error: no test specified\" && exit 1"  
},
```



```
"scripts": {  
  "test": "jest"  
},
```

```
▼ CALC  
  > node_modules  
  {} package-lock.json  
  {} package.json
```

```
{ } package.json ×  
{ } package.json > ...  
1  {  
2    "devDependencies": {  
3      "jest": "^29.7.0"  
4    },  
5    "name": "calc",  
6    "version": "1.0.0",  
7    "main": "index.js",  
8    "dependencies": {  
9      "ansi-escapes": "^4.3.2",  
10     "ansi-regex": "^5.0.1",
```

# Jest – test kalkulatora

// stwórz plik *index.js* i napisz funkcję *count(a,b,op)*:

```
JS index.js  X
JS index.js > ...
1  // JSDoc
2  /**
3   * funkcja wykonuje podstawowe działania na liczbach a i b oraz zwraca wynik
4   *
5   * @param {number} a - liczba a
6   * @param {number} b - liczba b
7   * @param {string} op - operator (+ - * /)
8   * @returns {string} - wynik obliczeń
9   */
10 > function count(a,b,op){ ...
31 }
32
33 // export modułu
34 module.exports = count;
```

# Jest – test kalkulatora

// stwórz plik *index.test.js* i napisz testy funkcji *count*:

```
JS index.test.js ×  
JS index.test.js > ...  
1 // import modułu  
2 const count = require('./index');  
3  
4 > test('1 + 2 = 3', () => { ...  
6   });
```

# Jest – test kalkulatora

// uruchom testy:

// *npm test*

```
● PS D:\calc> npm test

> calc@1.0.0 test
> jest

PASS ./index.test.js
  ✓ 1 + 2 = 3 (4 ms)
  ✓ 12 - 2 = 10
  ✓ 10 * 2 = 20 (1 ms)
  ✓ 10 / 2 = 5
  ✓ 10 / 0 = nie dzielimy przez zero

Test Suites: 1 passed, 1 total
Tests:       5 passed, 5 total
Snapshots:   0 total
Time:        3.096 s
Ran all test suites.
```

# Jest – test kalkulatora

// wykonaj dokumentację funkcji *count(a,b,op)*:

```
/function count(a,b,op){
```

```
/*function count(a,b,op){
```

```
/**function count(a,b,op){
```

```
 /** Autogenerated JSDoc */
```

Enter<-|

```
/**  
 * Description placeholder  
 *  
 * @param {number} a  
 * @param {number} b  
 * @param {string} op  
 * @returns {string}  
 */  
function count(a,b,op){
```



# Jest – test kalkulatora

// wygeneruj dokumentację funkcji *count(a,b,op)*:  
*jsdoc index.js*

## Global

Home

Global

count

## Methods

`count(a, b, op) → {string}`

funkcja wykonuje podstawowe działania na liczbach a i b oraz zwraca wynik

### Parameters:

Name	Type	Description
a	number	liczba a
b	number	liczba b
op	string	operator (+ - * /)

Source: [index.js, line 10](#)

### Returns:

- wynik obliczeń

Type

string

# JSDoc - an API documentation generator for JavaScript

@use JSDoc

## Getting Started with JSDoc 3

### Getting started

JSDoc 3 is an API documentation generator for JavaScript, similar to Javadoc or phpDocumentor. tool will scan your source code and generate an HTML documentation website for you.

### Adding documentation comments to your code

JSDoc's purpose is to document the API of your JavaScript application or library. It is assumed the so on.

# JSDoc

tagi używane w generatorze dokumentacji

## Block tags

[@abstract](#) (synonyms: [@virtual](#))

This member must be implemented (or overridden) by the inheritor.

[@access](#)

Specify the access level of this member (private, package-private, public, or protected).

[@alias](#)

Treat a member as if it had a different name.

[@async](#)

Indicate that a function is asynchronous.

[@augments](#) (synonyms: [@extends](#))

Indicate that a symbol inherits from, and adds to, a parent symbol.

[@author](#)

Identify the author of an item.

[@borrows](#)

This object uses something from another object.

[@class](#) (synonyms: [@constructor](#))

This function is intended to be called with the "new" keyword.

[@classdesc](#)

Use the following text to describe the entire class.

[@constant](#) (synonyms: [@const](#))

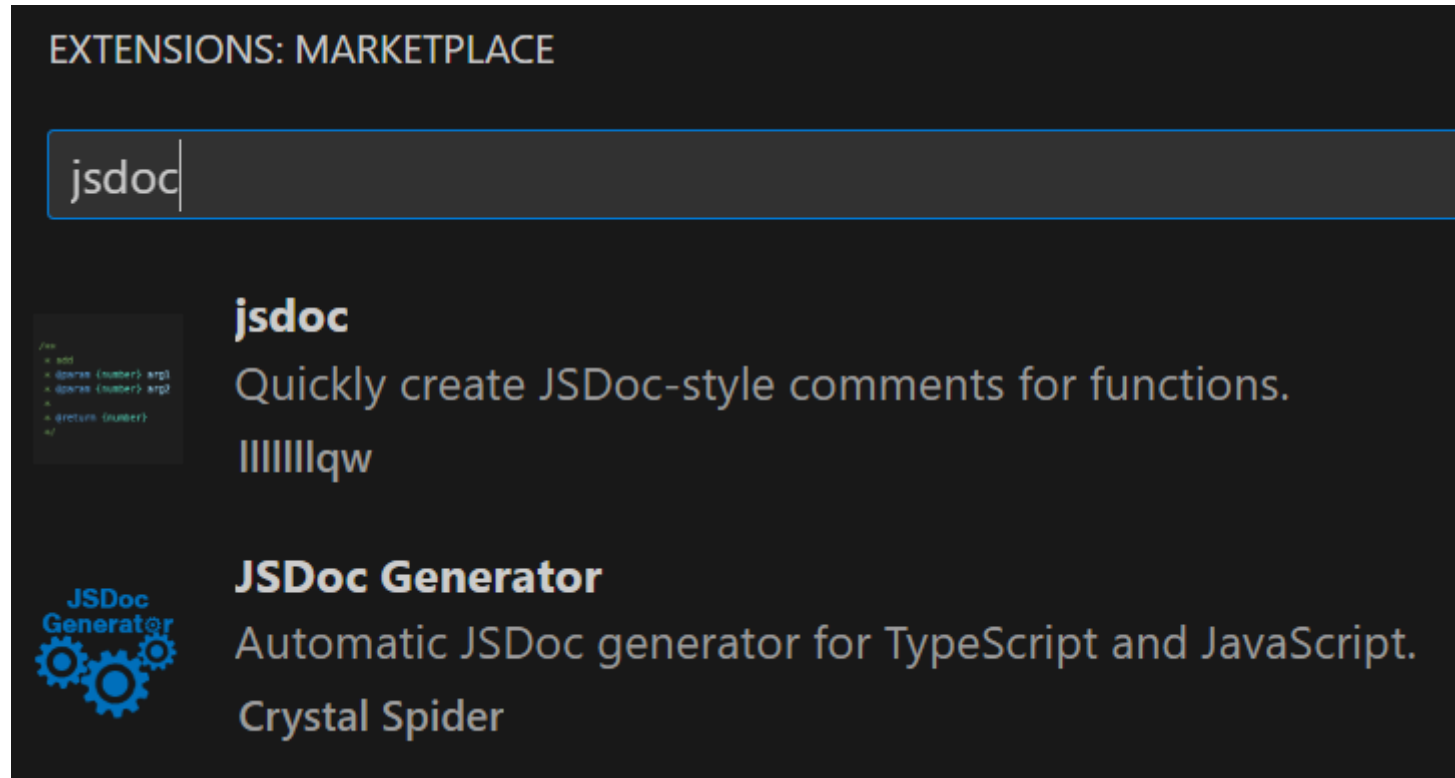
Document an object as a constant.

# JSDoc – Visual Studio Code

```
1  /**
2   * Documentation
3   * @param {number} a Some number
4   * @param {number} b Same
5   */
6  function mul(a, b) {
7    ...return a * b
8  }
```

[https://code.visualstudio.com/Docs/languages/javascript#\\_jsdoc-support](https://code.visualstudio.com/Docs/languages/javascript#_jsdoc-support)

# JSDoc – Visual Studio Code



należy zainstalować w Visual Studio Code wsparcie dla JSDoc

# JSDoc – Visual Studio Code - Hover Information

```
/**
 * Documentation
 * @description the function calculates the product of 2 numbers
 * @param {number} a some number
 * @param {number} b some number
 * @returns multiplication a*b
 */
function mul(a,b){
  return a * b
}
```

```
function mul(a: number, b: number): number

Documentation
@description — the function calculates the product of 2 numbers
@param a — some number
@param b — some number
@returns — multiplication a*b

/**
 * Docume
 * @descr
 * @param
 * @param
 * @retur
 */
function mul(a,b){
  return a * b
}
```

# JSDoc – Visual Studio Code – doc generator

```
npm install -g jsdoc
```

komenda instalująca generator dokumentacji

gdy w ścieżce pliku pojawi się znak podkreślenia dokumentacja się nie wygeneruje !

By default, JSDoc ignores files and **directories** that start with an underscore (\_).

<https://stackoverflow.com/questions/24412723/the-error-message-there-are-no-input-files-to-process-from-jsdoc>

# JSDoc – Visual Studio Code – doc generator

main.js

```
/** @description klasa opisująca salę lekcyjną komputerową
 *
 */
class Sala {
  numer;
  szerokosc;
  dlugosc;
  rzutnik;
  koszNaSmieci;
  iloscKomputerow;

  /**
   * @description konstruktor
   * @param {Number} num numer sali
   * @param {Number} szer szerokość sali
   * @param {Number} dl dlugosc sali
   * @param {Boolean} rzut czy jest rzutnik
   * @param {Boolean} kosz czy jest kosz na śmieci
   * @param {Number} ilosc ilość komputerów
   */
  constructor(num, szer, dl, rzut, kosz, ilosc){
    this.numer = num;
    this.szerokosc = szer;
    this.dlugosc = dl;
    this.rzutnik = rzut;
    this.koszNaSmieci = kosz;
    this.iloscKomputerow = ilosc;
  }
  /**
   * @description powierzchnia sali
   */
}
```

jsdoc main.js

Nazwa

out

index.html

main.js

Class: Sala

Sala(num, szer, dl, rzut, kosz, ilosc)

new Sala(num, szer, dl, rzut, kosz, ilosc)

klasa opisująca salę lekcyjną komputerową

Parameters:

Name	Type	Description
num	Number	numer sali
szer	Number	szerokość sali
dl	Number	dlugosc sali
rzut	Boolean	czy jest rzutnik
kosz	Boolean	czy jest kosz na śmieci
ilosc	Number	ilość komputerów

Source: [main.js, line 4](#)



# doxygen

The screenshot shows the Doxygen website interface. At the top, there is a navigation bar with links for 'Docs', 'Changelog', 'Extensions', and 'Examples'. A 'Download' button and a 'Donate' button are also present. A banner below the navigation bar announces 'Version 1.12.0 is now available! Release date: 7 August 2024'. The main content area displays the 'Doxygen' logo and a search bar. On the left, a sidebar lists various classes and methods, with 'Cache' expanded to show methods like 'const\_iterator', 'iterator', 'kv\_pair', 'Cache', 'begin', 'capacity', 'clear', 'end', 'find', 'hits', 'insert', 'misses', and 'remove'. The main panel shows the documentation for the 'find()' method, including its signature: `template<typename K, typename V > V * Cache< K, V >::find ( const K & key )`. The description states: 'Finds a value in the cache given the corresponding key.' The 'Returns' section indicates: 'a pointer to the value or nullptr if the key is not found in the cache'. A 'Note' box contains the text: 'The hit and miss counters are updated, see hits() and misses()'. The 'Definition at line 105 of file cache.h.' is followed by a code block showing the implementation of the method. The code includes comments and logic for moving the item to the front of the list and returning the value or nullptr. The 'References' section at the bottom lists: 'References Cache< K, V >::m\_cacheItemList, Cache< K, V >::m\_cacheItemMap, Cache< K, V >::m\_hits, and Cache< K, V >::m\_misses.'

Code  
Documentation.  
Automated.

Free, open source, cross-platform.

<https://www.doxygen.nl/>

# słowniczek

**SDLC (*Software Development Lifecycle*)** to proces wykorzystywany do opracowywania oprogramowania, które spełnia oczekiwania klientów, a także uwzględnia ograniczenia budżetowe i czasowe. <https://www.ovhcloud.com/pl/learn/what-is-sdlc/>

Tworzenie **przypadków użycia (*use case*)** – technika stosowana w inżynierii oprogramowania w celu opisanie wymagań tworzonego systemu informatycznego. Przypadek użycia przedstawia interakcję pomiędzy aktorem (użytkownikiem systemu), który inicjuje zdarzenie oraz samym systemem jako sekwencję prostych kroków. [https://pl.wikipedia.org/wiki/Przypadek u%C5%BCycia](https://pl.wikipedia.org/wiki/Przypadek_u%C5%BCycia)

**Use case diagram** [https://en.wikipedia.org/wiki/Use\\_case\\_diagram](https://en.wikipedia.org/wiki/Use_case_diagram)

**Historyjka użytkownika (*user story*<sup>l</sup>)** – metoda opisywania wymagań przy tworzeniu oprogramowania w metodykach zwinnych. Historyjki są zazwyczaj pisane z perspektywy użytkownika końcowego. Stosowany jest prosty język, który jest zrozumiały zarówno przez programistów, jak i osoby nietechniczne (np. analitycy, projektanci, menedżerowie). [https://pl.wikipedia.org/wiki/Historyjka u%C5%BCytkownika](https://pl.wikipedia.org/wiki/Historyjka_u%C5%BCytkownika)

**linki**

<https://www.toolsqa.com/>