Your First VS Code Extension

Andrei Voronin

About

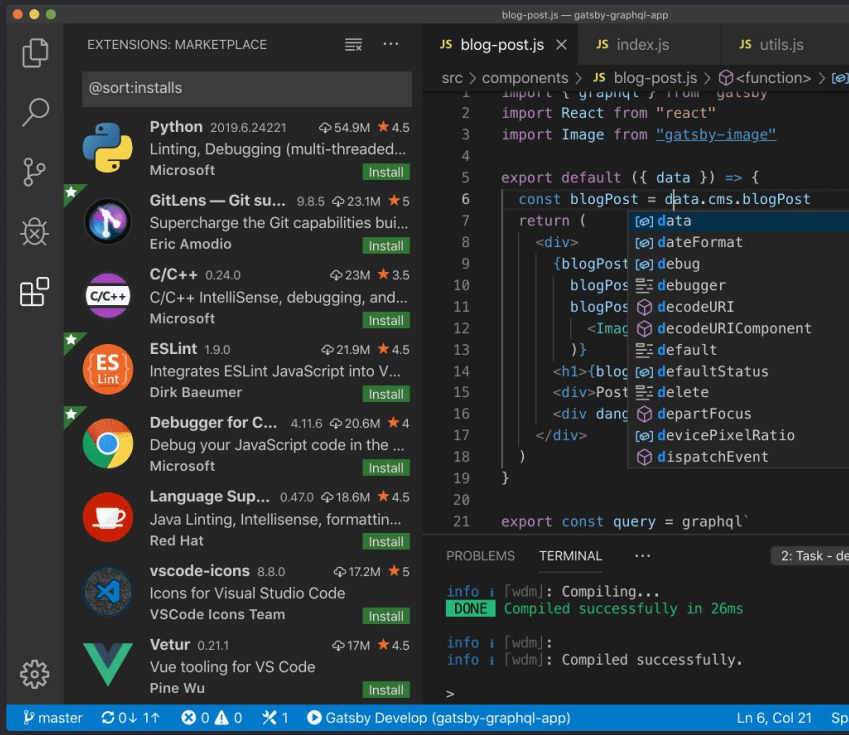
- Ecommerce CMS
- Frontend
- Storefront &
  Admin panel UX/UI

Accelerate development

Accelerate development

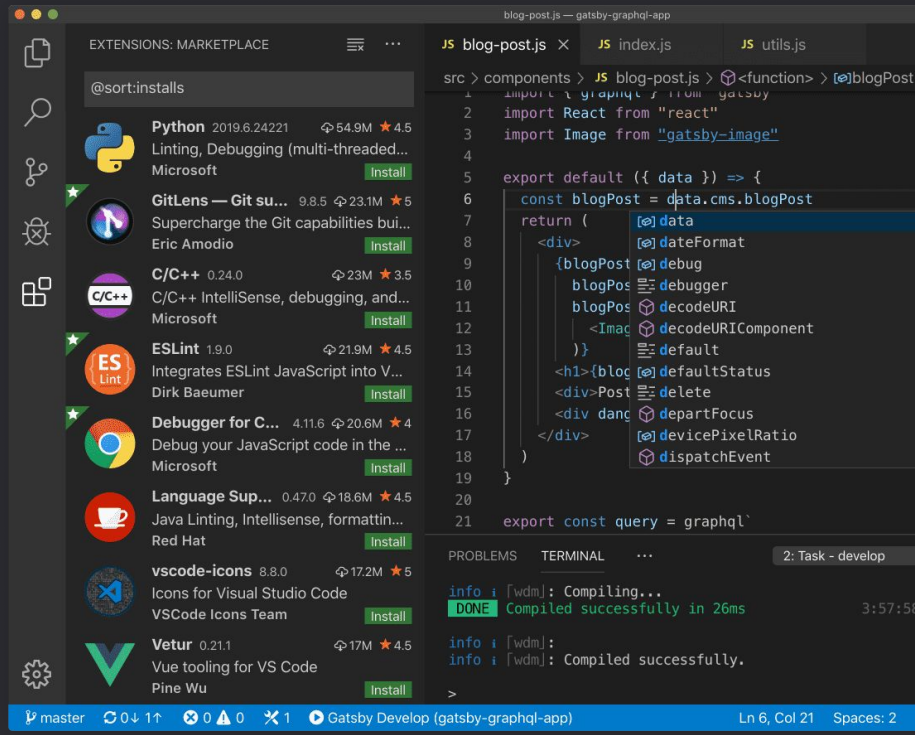
- Command palette
- Custom shortcuts/snippets
- Extensions
- Custom extensions

Visual Studio Code



Visual Studio Code

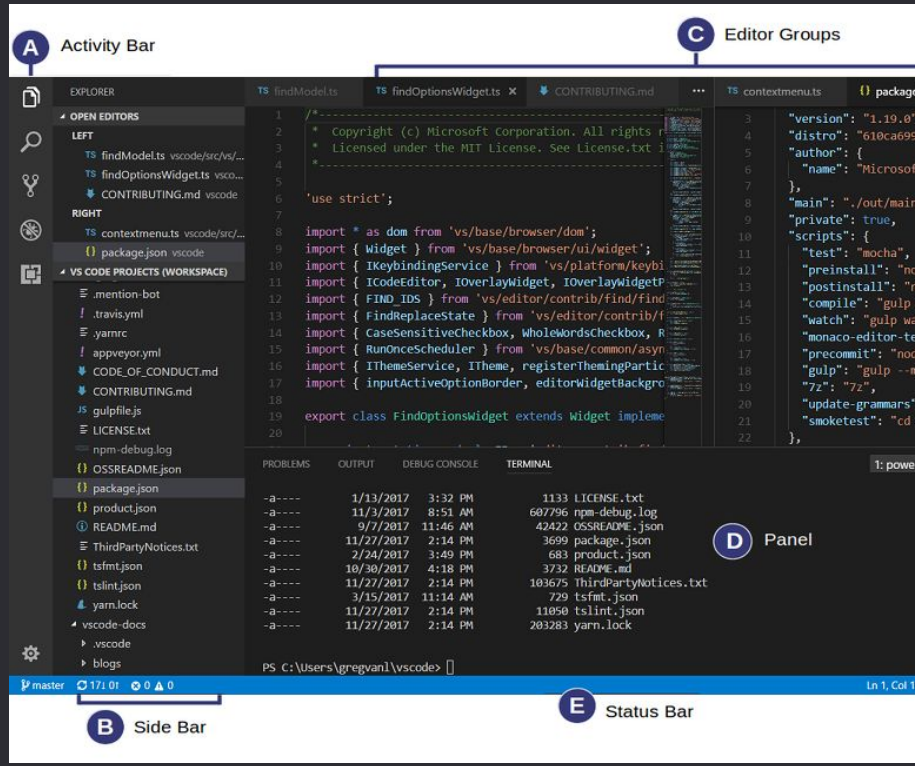
- Fast & free
- Built-in Git + GitLens
- Actively developing



code.visualstudio.com

Built with extensibility in mind

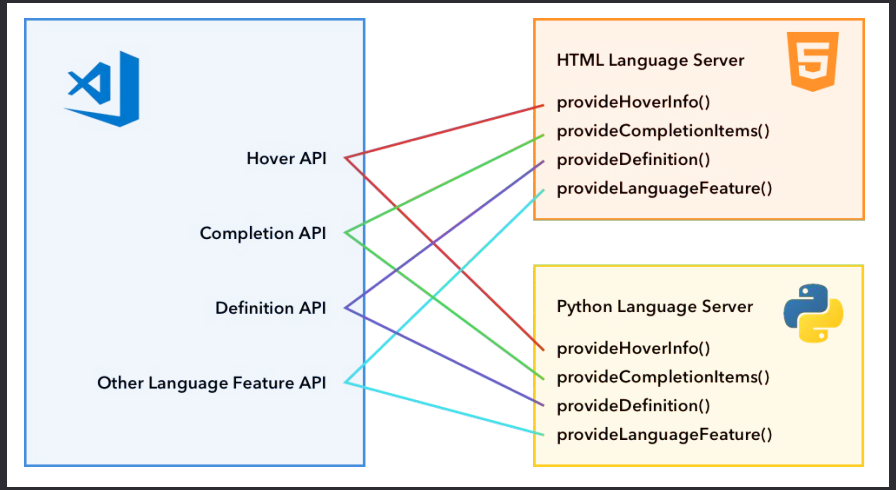
- Fully customizable
- Core features as extensions
- Like browser extensions



code.visualstudio.com/api

Capabilities

- Syntax highlighting
- Hover info & completion
- Jump to definition
- Error checking
- Shortcut & context menu
- Notification & status bar info
- Activity bar view



code.visualstudio.com/api/extension-capabilities/overview

Hello world

Dev environment

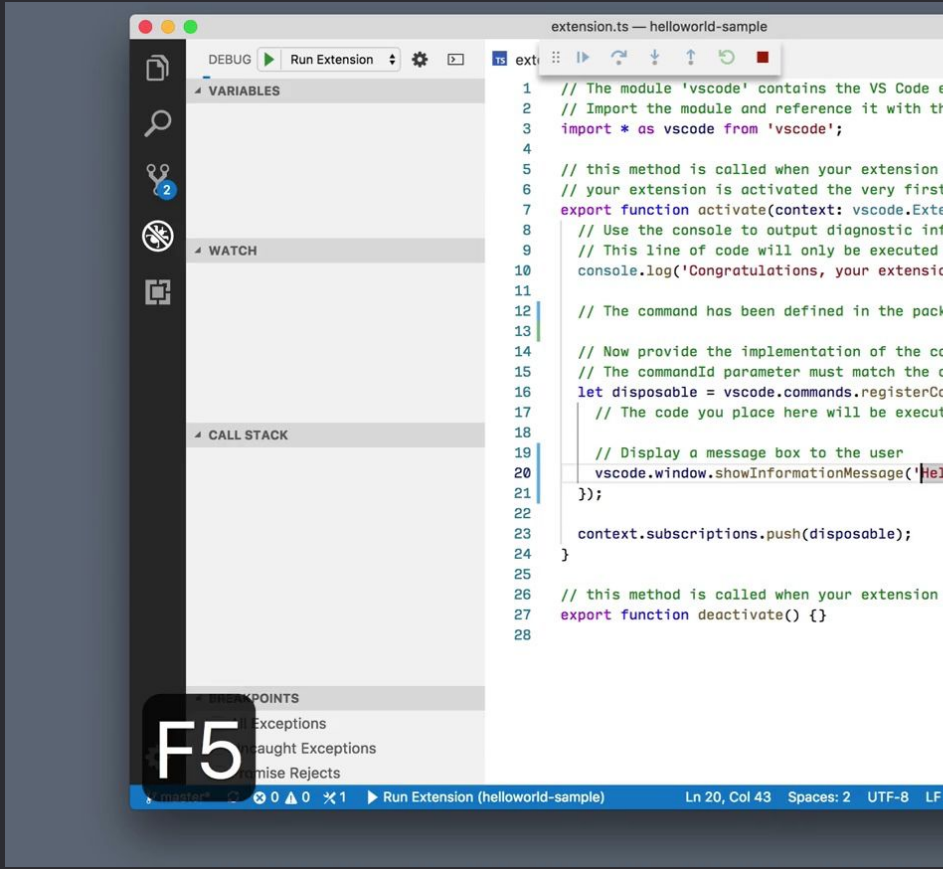
- yarn global add
  yo generator-code
- yo code
  - New Extension (TypeScript)
- code ./helloworld

```
# ? What type of extension do you want to create?
    New Extension (TypeScript)
# ? What's the name of your extension?
    HelloWorld
    ### Press <Enter> to choose default
    for all options below ###

# ? What's the identifier of your extension?
    helloworld
# ? What's the description of your extension?
    LEAVE BLANK
# ? Initialize a git repository?
    Yes
# ? Which package manager to use?
    yarn
```

code.visualstudio.com/api/get-started/your-first-extension

Run

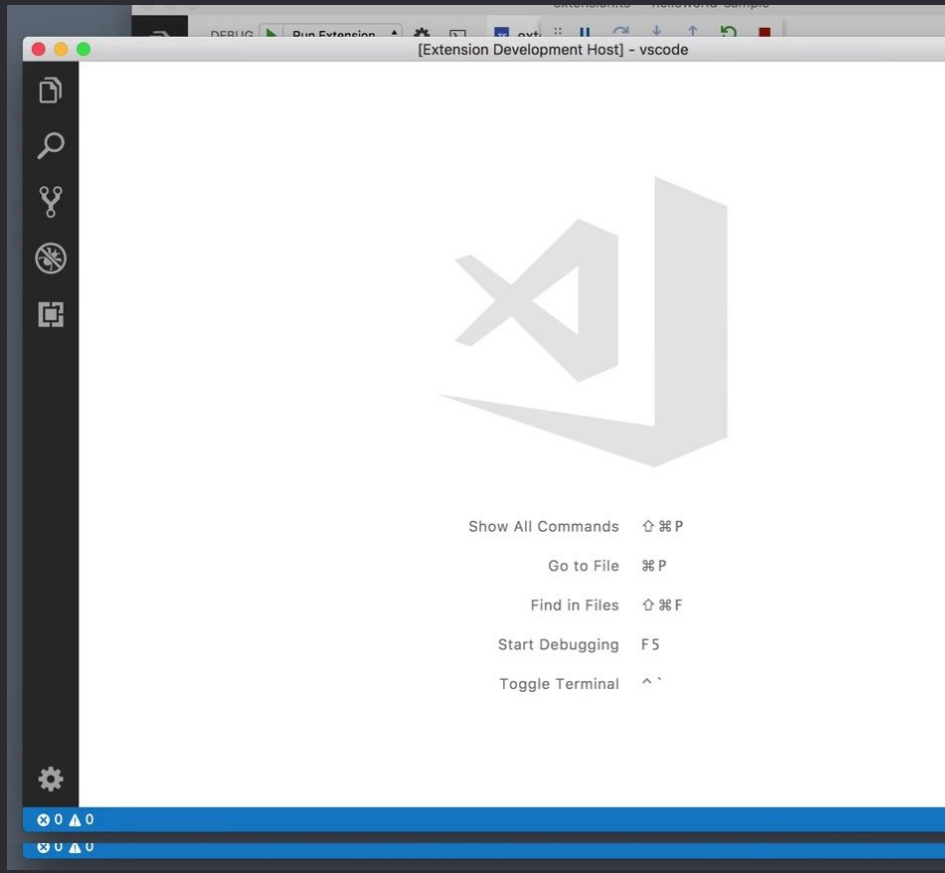
1. Start debugging F5
2. Extension Development Host window
3. Command Palette ⇧⌘P
4. Run the Hello World
5. Notification showing up



code.visualstudio.com/api/get-started/your-first-extension

Run

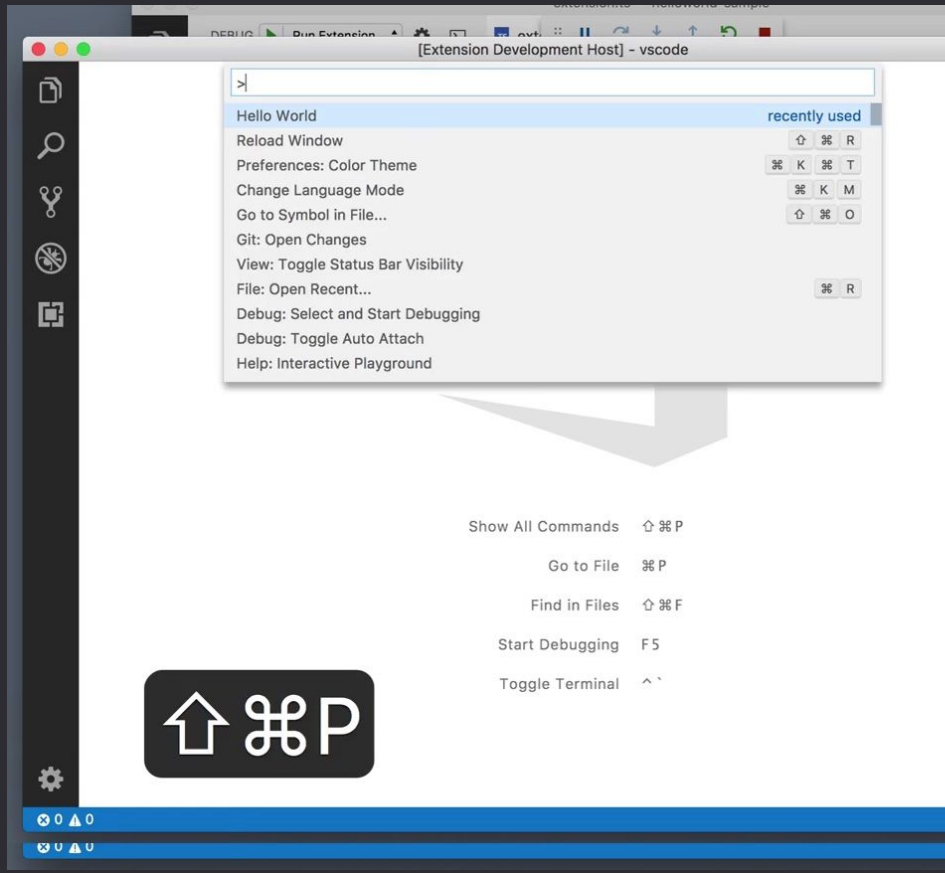
1. Start debugging F5
2. Extension Development Host window
3. Command Palette ⇧⌘P
4. Run the Hello World
5. Notification showing up



code.visualstudio.com/api/get-started/your-first-extension

Run

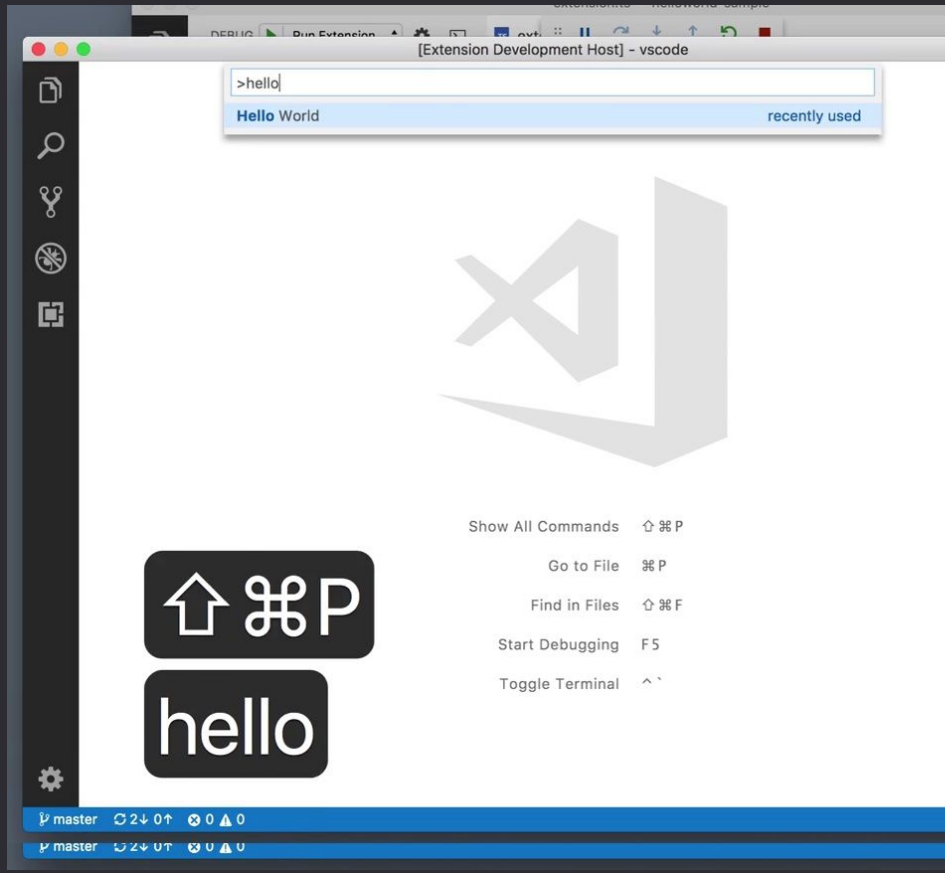
1. Start debugging F5
2. Extension Development Host window
3. Command Palette ⇧⌘P
4. Run the Hello World
5. Notification showing up



code.visualstudio.com/api/get-started/your-first-extension

Run

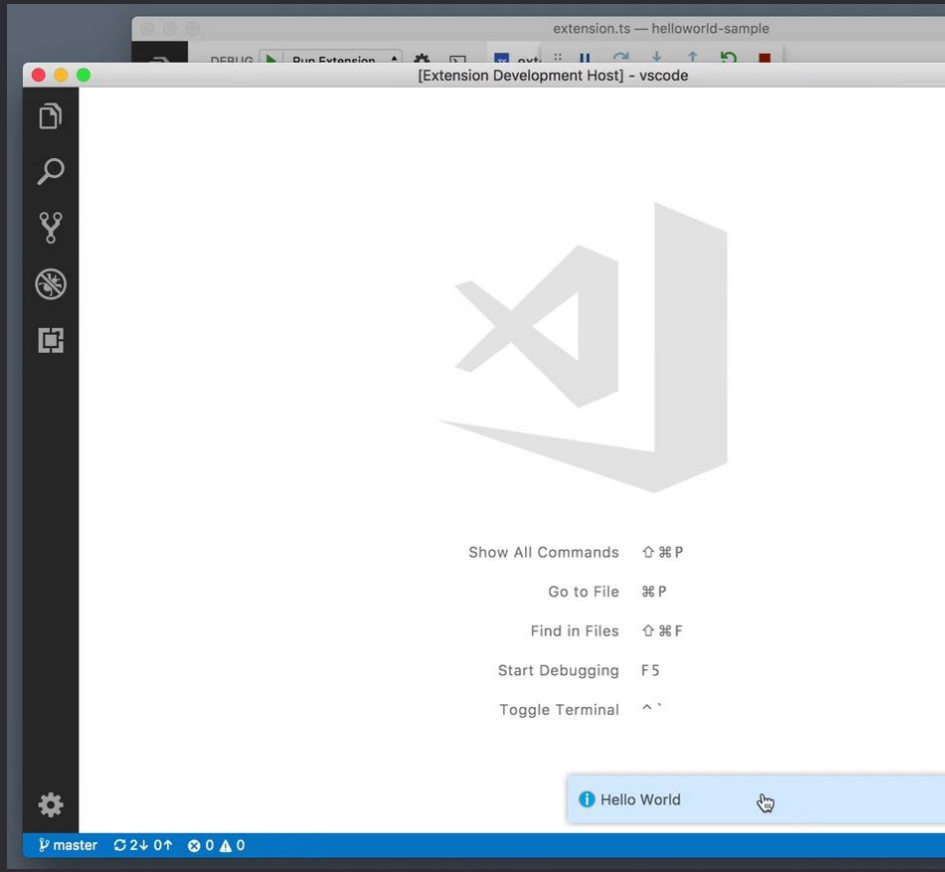
1. Start debugging F5
2. Extension Development Host window
3. Command Palette ⇧⌘P
4. Run the Hello World
5. Notification showing up



code.visualstudio.com/api/get-started/your-first-extension

Run

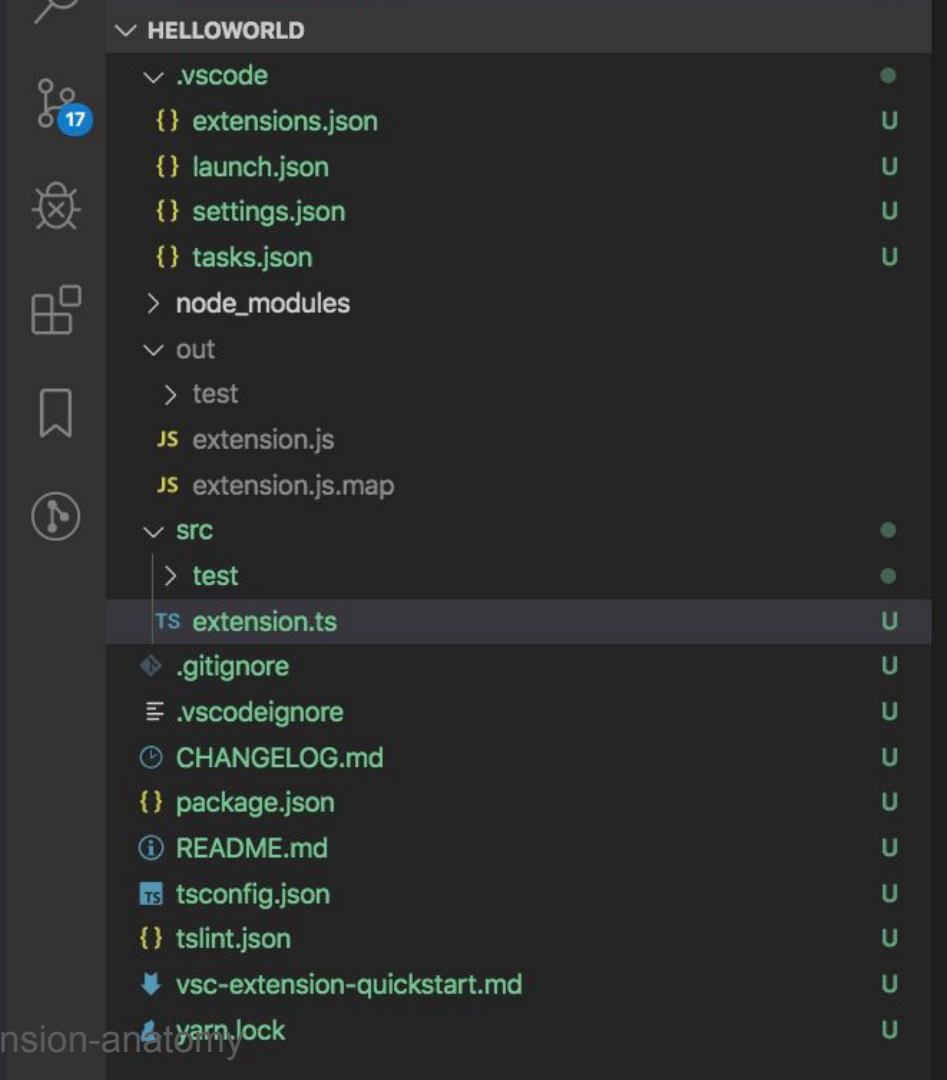
1. Start debugging F5
2. Extension Development Host window
3. Command Palette ⇧⌘P
4. Run the Hello World
5. Notification showing up



code.visualstudio.com/api/get-started/your-first-extension

Structure

- Extension Manifest
  - package.json
- Extension Entry File
  - extensions.ts



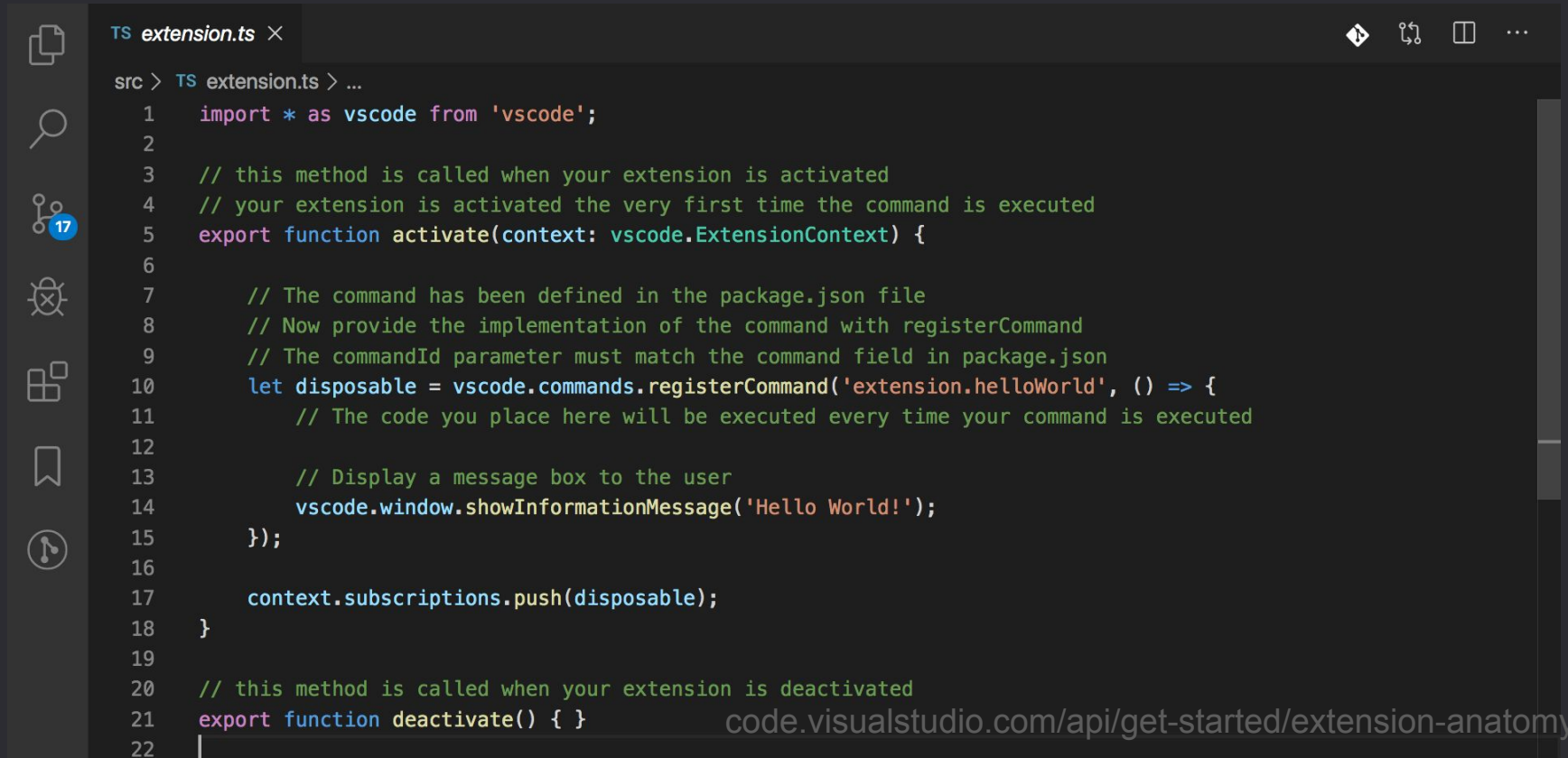
code.visualstudio.com/api/get-started/extension-anatomy

Manifest package.json

```
{
  "name": "helloworld-sample",
  "publisher": "vscode-samples",
  "main": "./out/extension.js",
  "activationEvents": ["onCommand:extension.helloWorld"],  // ["*"]
  "contributes": {
    "commands": [
      {
        "command": "extension.helloWorld",
        "title": "Hello World"
      }
    ]
  },
}
```

code.visualstudio.com/api/get-started/extension-anatomy

Entry File extensions.ts



code.visualstudio.com/api/get-started/extension-anatomy

Anatomy

1.

Registers the `onCommand`

Activation Event

package.json

2.

Uses the `contributes.commands`

Contribution Point

package.json

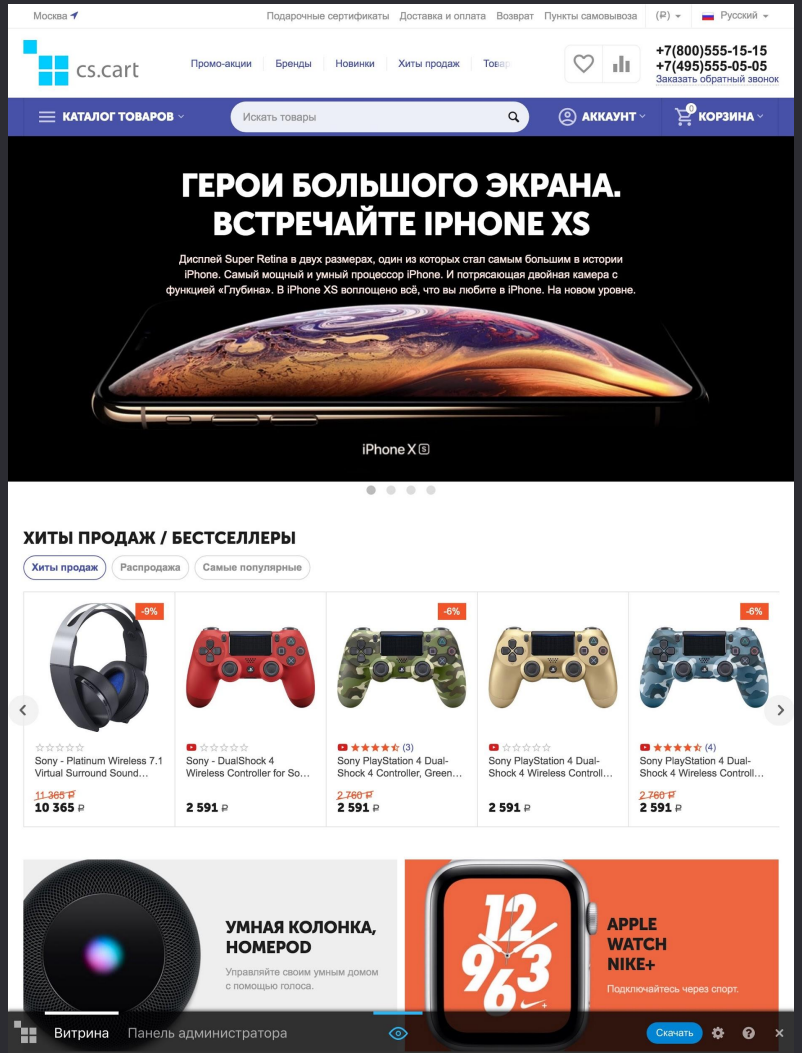
3.

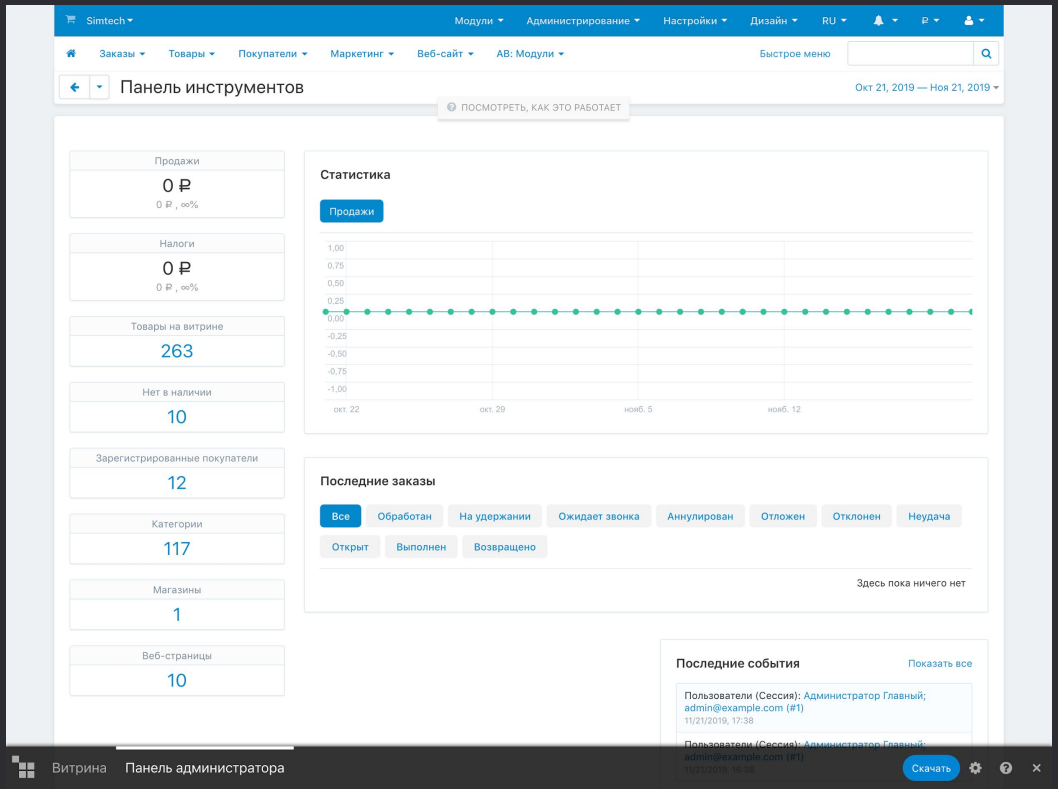
Uses the `commands.registerCommand`

VS Code API

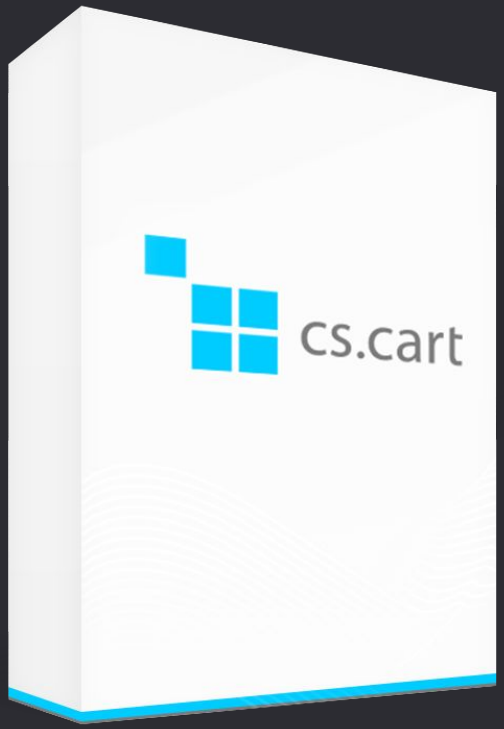
extensions.ts

code.visualstudio.com/api/get-started/extension-anatomy


Москва
Подарочные сертификаты
Доставка и оплата
Возврат
Пункты самовывоза
(₽)
Русский
cs.cart
Промо-акции
Бренды
Новинки
Хиты продаж
+7(800)555-15-15
+7(495)555-05-05
Заказать обратный звонок
КАТАЛОГ ТОВАРОВ
Искать товары
АККАУНТ
КОРЗИНА
ГЕРОИ БОЛЬШОГО ЭКРАНА.
ВСТРЕЧАЙТЕ IPHONE XS
Дисплей Super Retina в двух размерах, один из которых стал самым большим в истории iPhone. Самый мощный и умный процессор iPhone. И потрясающая двойная камера с функцией «Глубина». В iPhone XS воплощено всё, что вы любите в iPhone. На новом уровне.
iPhone XS
ХИТЫ ПРОДАЖ / БЕСТСЕЛЛЕРЫ
Хиты продаж
Распродажа
Самые популярные
-9%
Sony - Platinum Wireless 7.1 Virtual Surround Sound…
11 365 ₽
10 365 ₽
Sony - DualShock 4 Wireless Controller for So…
2 591 ₽
-6%
Sony PlayStation 4 Dual-Shock 4 Controller, Green…
2 760 ₽
2 591 ₽
Sony PlayStation 4 Dual-Shock 4 Wireless Controll…
2 591 ₽
-6%
(4)
Sony PlayStation 4 Dual-Shock 4 Wireless Controll…
2 760 ₽
2 591 ₽
УМНАЯ КОЛОНКА, HOMEPOD
Управляйте своим умным домом с помощью голоса.
APPLE WATCH NIKE+
Подключайтесь через спорт.
Витрина
Панель администратора
Скачать


Simtech
Модули
Администрирование
Настройки
Дизайн
RU
Заказы
Товары
Покупатели
Маркетинг
Веб-сайт
АВ: Модули
Быстрое меню
Панель инструментов
Окт 21, 2019 — Ноя 21, 2019
ПОСМОТРЕТЬ, КАК ЭТО РАБОТАЕТ
Продажи
0 ₽
0 ₽ , ∞%
Налоги
0 ₽
0 ₽ , ∞%
Товары на витрине
263
Нет в наличии
10
Зарегистрированные покупатели
12
Категории
117
Магазины
1
Веб-страницы
10
Статистика
Продажи
1,00
0,75
0,50
0,25
0,00
-0,25
-0,50
-0,75
-1,00
окт. 22
окт. 29
нояб. 5
нояб. 12
Последние заказы
Все
Обработан
На удержании
Ожидает звонка
Аннулирован
Отложен
Отклонен
Неудача
Открыт
Выполнен
Возвращено
Здесь пока ничего нет
Последние события
Показать все
Пользователи (Сессия): Администратор Главный; admin@example.com (#1)
11/21/2019, 17:38
Витрина
Панель администратора
Скачать

CMS



Framework

github.com/andreivoronin/

cscartcode

Language variables

{__("language_variables")}

```
85      {dropdown content=$smarty.capture.tools_list}
86      </span>
87
88      {if $products}
89          {include file="buttons/save.tpl" but_name="dispatch[products.m_update]" but_role=
90      {/if}
91  {/captur
92
93  {capture
94
95  <p>{__("text_select_fields2edit_note")}</p>
96  {include file="views/products/components/products_select_fields.tpl"}
97
98  <div class="buttons-container">
99      {include file="buttons/save_cancel.tpl" but_text=__("modify_selected") but_name="disp
100 </div>
101 {/capture}
102 {include file="common/popupbox.tpl" id="select_fields_to_edit" text=__("select_fields_to_
103
104 </form>
105
106 {/capture}
```

```
msgctxt "Languages::text_select_fields2edit_note"
msgid "You are able to modify several items at once. Select the chec
k boxes corresponding to the fields you wish to edit, and click on M
odify Selected."
```

extension.ts

```
1  import * as vscode from 'vscode';
2  import { LanguageVariableDefinitionProvider } from './LanguageVariableDefinitionProvider';
3
4  export function activate(context: vscode.ExtensionContext) {
5
6      let languageVariableDefinitionProvider = vscode.languages.registerDefinitionProvider(
7          {
8              scheme: 'file',
9              language: 'smarty',
10         }
11         , new LanguageVariableDefinitionProvider());
12
13     context.subscriptions.push(languageVariableDefinitionProvider);
14 }
15
16 export function deactivate() {}
17
```

LanguageVariableDefinitionProvider.ts

- provideDefinition(document: TextDocument, position: Position, token: CancellationToken): Promise<Definition | undefined>
- /__\("[\w\.]+"\)/
- line = document.lineAt(position)
- new Location(Uri.file(workspaceFolders + poFilePath), new Position(langVarLineNumber,0))

LanguageVariableDefinitionProvider.ts

```
1   import fs = require('fs');
2   import {
3     CancellationToken,
4     TextDocument,
5     DefinitionProvider,
6     Definition,
7     Location,
8     Position,
9     workspace,
10    Uri,
11    TextLine
12  } from 'vscode';
13
14  const fsPromises = fs.promises;
15  const LANGVAR_MATCHER = /__\("[\w\.]+"\)/;
16  const START_LANGVAR_MATCHER = /__\("[\w\.]+/;
17  const START_LANGVAR_SYMBOL = '__("';
18  const END_LANGVAR_SYMBOL = '")';
19  const startLangvarSymbolLength = START_LANGVAR_SYMBOL.length;
20  const endLangvarSymbolLength = END_LANGVAR_SYMBOL.length;
21
22  let textBeforeLangVar = '';
23  let langVarLineNumber = 0;
24  let poFilePath = '/var/langs/en/core.po';
```

LanguageVariableDefinitionProvider.ts

```
26  export class LanguageVariableDefinitionProvider implements DefinitionProvider {
27
28    public async provideDefinition(document: TextDocument, position: Position,
29      token: CancellationToken): Promise<Definition | undefined> {
30      if (!workspace.workspaceFolders) {
31        return;
32      }
33
34      const line = document.lineAt(position);
35      const tag = getLangVarFromLine(line);
36
37      if (!tag) {
38        return;
39      }
40
41      const workspaceFolders = workspace.workspaceFolders[0].uri.path;
42      const tagDirtyText = tag ? tag[0] : '';
43      const tagText = tagDirtyText.substring(startLangvarSymbolLength,
44        tagDirtyText.length - endLangvarSymbolLength);
45      const langvarPosition = getLangVarPosition(tag);
```

LanguageVariableDefinitionProvider.ts

```
46
47        if (position.character < langvarPosition.start || position.character > langvarPosition.end){
48          return;
49        }
50
51        const fileText = await readFile(workspaceFolders, poFilePath) || '';
52        const langVarIndex = fileText.indexOf(tagText);
53
54        if (langVarIndex > 0) {
55          textBeforeLangVar = fileText.toString().substring(0, langVarIndex);
56          langVarLineNumber = textBeforeLangVar.split(/\r?\n/).length - 1;
57        }
58
59        return new Location(Uri.file(workspaceFolders + poFilePath), new Position(langVarLineNumber,0)
60      }
61    }
```

```
63  async function readFile(workspaceFolders: string, poFilePath: string) {
64    try {
65      return fsPromises.readFile(workspaceFolders + poFilePath);
66    } catch (err) {
67      console.error('Error occured while reading file!', err);
68    }
69  }
70
71  function getLangVarFromLine(line: TextLine) {
72    const tagStart = START_LANGVAR_MATCHER.exec(line.text);
73    const tagStartIndex = tagStart ? tagStart.index : 0;
74
75    return LANGVAR_MATCHER.exec(line.text.substring(tagStartIndex));
76  }
77
78  function getLangVarPosition(tag: RegExpExecArray | null) {
79    const start = tag ? tag.index + startLangvarSymbolLength + 1 : 0;
80    const tagLength = tag ? tag[0].length - endLangvarSymbolLength : 0;
81    const end = start + tagLength;
82
83    return { start, end, };
84  }
```

Include

{include file="template.tpl"}

```
577         {if $id}
578             {** Product options section **}
579             <div class="cm-hide-save-button hidden" id="content_options">
580                 {include file="views/products/components/products_update_options.tpl"}
581             </div>
582             {** /Product options section **}
583 
584             {** Products files section **}
585             {if $settings.General.enable_edp == "Y"}
586             <div id="content_files" class="cm-hide-save-button hidden">
587                 {hook name="products:content_files"}
588                 {include file="views/products/components/products_update_files.tpl"}
589                 {/hook}
590             </div>
591             {/if}
592             {** /Products files section **}
593 
594             {** Subscribers section **}
595             <div id="content_subscribers" class="cm-hide-save-button hidden">
596                 {include file="views/products/components/product_subscribers.tpl" product_id=$id}
597             </div>
598             {** /Subscribers section **}
599         {/if}
600 
601     {/capture}
602     {include file="common/tabsbox.tpl" content=$smarty.capture.tabsbox group_name=$runtime.controller active_t
603 
```

extension.ts

```
1   import * as vscode from 'vscode';
2   import { IncludeLinkProvider } from './IncludeLinkProvider';
3
4   export function activate(context: vscode.ExtensionContext) {
5
6       let disposable = vscode.languages.registerDocumentLinkProvider(
7           {
8               scheme: 'file',
9               language: 'smarty',
10          }
11          , new IncludeLinkProvider());
12
13      context.subscriptions.push(disposable);
14  }
```

IncludeLinkProvider.ts

- `provideDocumentLinks(document: TextDocument, token: CancellationToken): Promise<DocumentLink[]>`
- `/{include/g, /file=\"\S*\.tpl\"/g`
- `line = document.lineAt(l);`
- `Position(line.lineNumber, fileMatchesLine.index + 6)`
- `new DocumentLink(new Range(start, end), url)`

IncludeLinkProvider.ts

```ts
import * as path from 'path';
import fs = require('fs');
import * as vscode from 'vscode';
import {
  CancellationToken,
  DocumentLink,
  DocumentLinkProvider,
  TextDocument,
  workspace
} from 'vscode';
import {
  Position,
  Range,
  Uri
} from 'vscode';

const INCLUDE_MATCHER = /{include/g;
const FILE_MATCHER = /file=\"\S*\.tpl\"/g;

export class IncludeLinkProvider implements DocumentLinkProvider {
  public async provideDocumentLinks(document: TextDocument, token: CancellationToken): Promise<DocumentLink[]> {
    const urls = await findIncludeUrl(document);
    return Promise.resolve(urls ? urls : []);
  }
}
```

IncludeLinkProvider.ts

```
async function findIncludeUrl(document: TextDocument): Promise<DocumentLink[] | undefined> {
  const results: DocumentLink[] = [];
  let line: vscode.TextLine;
  let searchString: string = '';
  let foundMultiline: Boolean = false;
  let documentLink: DocumentLink | undefined;

  for (let l = 0; l < document.lineCount; l++) {
    line = document.lineAt(l);

    if ((INCLUDE_MATCHER.exec(line.text)) !== null && !line.text.includes('}')) {
      searchString = line.text;
      foundMultiline = true;

      documentLink = await getLink(line);
      if (documentLink) {
        results.push(documentLink);
      }

      continue;

    } else if (foundMultiline && !line.text.includes('}')) {
      searchString += line.text;

      documentLink = await getLink(line);
      if (documentLink) {
        results.push(documentLink);
```

IncludeLinkProvider.ts

```
54        }
55
56        continue;
57      } else if (foundMultiline && line.text.includes('}')) {
58        searchString += line.text;
59        foundMultiline = false;
60
61        documentLink = await getLink(line);
62        if (documentLink) {
63          results.push(documentLink);
64        }
65
66      } else {
67        searchString = line.text;
68
69        documentLink = await getLink(line);
70        if (documentLink) {
71          results.push(documentLink);
72        }
73      }
74    }
75
76    return results;
77  }
```

IncludeLinkProvider.ts

```
79  async function getLink(line: vscode.TextLine): Promise<DocumentLink | undefined> {
80    let fileMatchesLine: RegExpExecArray | null;
81    let link: string = '';
82    let url: Uri | undefined;
83    let start: Position = new Position(0, 0);
84    let end: Position = new Position(0, 0);
85
86    while ((fileMatchesLine = FILE_MATCHER.exec(line.text)) !== null) {
87      if (fileMatchesLine) {
88        link = fileMatchesLine[0].substring(6, fileMatchesLine[0].length - 1);
89        url = await getFileUrl(link);
90
91        start = new Position(line.lineNumber, fileMatchesLine.index + 6);
92        end = start.translate(0, link.length);
93
94        return new DocumentLink(new Range(start, end), url);
95      }
96    }
97  }
98
```

IncludeLinkProvider.ts

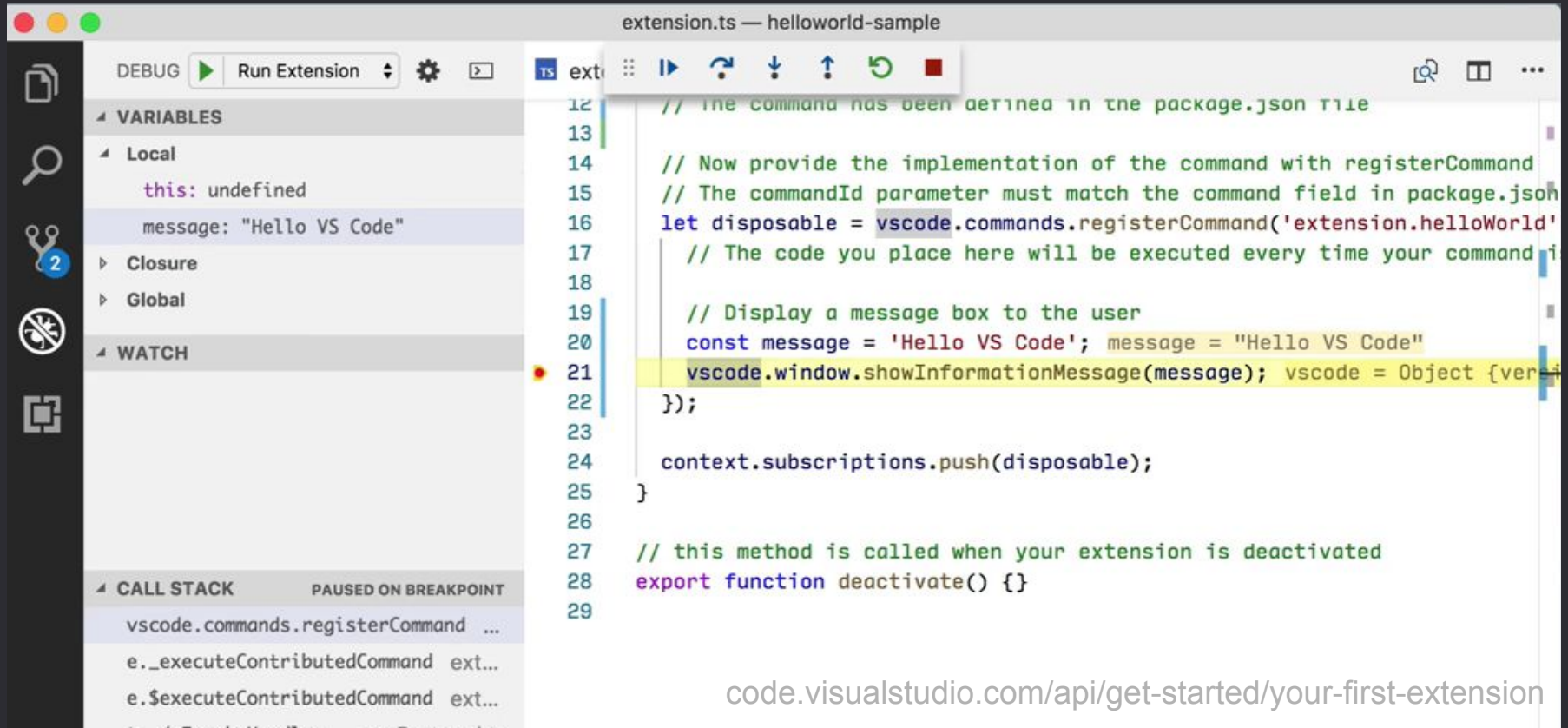
```
async function getFileUrl(link: string): Promise<Uri | undefined> {
  let validFilePath = await getValidFilePath(link);

  if (!validFilePath) {
    return;
  }

  return Uri.file(validFilePath);
}

async function getValidFilePath(link: string): Promise<string | undefined> {
  let filePath = await getFilePath(link);

  if (!filePath || !workspace.workspaceFolders) {
    return;
  }

  return fs.existsSync(filePath) ?
    filePath : [workspace.workspaceFolders[0].uri.path, 'design', 'backend', 'responsive', 'templates', link]
      .filter(entry => entry.trim() !== '')
      .join('/');
}
```

```
async function getFilePath(link: string): Promise<string | undefined> {
  if (!vscode.window.activeTextEditor) {
    await vscode.window.activeTextEditor;
  }

  if (!workspace.workspaceFolders) {
    return;
  }
  if (!vscode.window.activeTextEditor) {
    return;
  }
  let folderPath = '';

  if (vscode.window.activeTextEditor) {
    folderPath = workspace.asRelativePath(path.dirname(vscode.window.activeTextEditor.document.fileName));
  }

  let location = 'backend';
  let theme = '';

  if (folderPath.split('/')[1] === 'themes') {
    location = 'themes';
    theme = folderPath.split('/')[2];
  }

  return [workspace.workspaceFolders[0].uri.path, 'design', location, theme, 'templates', link]
    .filter(entry => entry.trim() !== '')
    .join('/');
}
```

Plans for Capabilities

- Snippets
- Hooks
- Show all available parameters
- Go to style
- Go to script plugin
- Core ↔ Addons

Debugging



code.visualstudio.com/api/get-started/your-first-extension

Tips

- VSCode API docs
- Explore other repositories
- Explore extensions

Packaging

Package preparation

1. yarn global add vsce
2. package.json:

```
"publisher": "andreivoronin",
"license": "SEE LICENSE IN LICENSE.txt",
"repository": {
  "type": "git",
  "url": "https://github.com/andreivoronin/cscartcode.git"
}
```

ERROR Missing publisher name.

warning package.json: No license field

WARNING A 'repository' field is missing

3. README.md: ERROR Make sure to edit the README.md file before you package...

~~This is the README for your extension~~

code.visualstudio.com/api/working-with-extensions/publishing-extension

Packaging

- vsce package

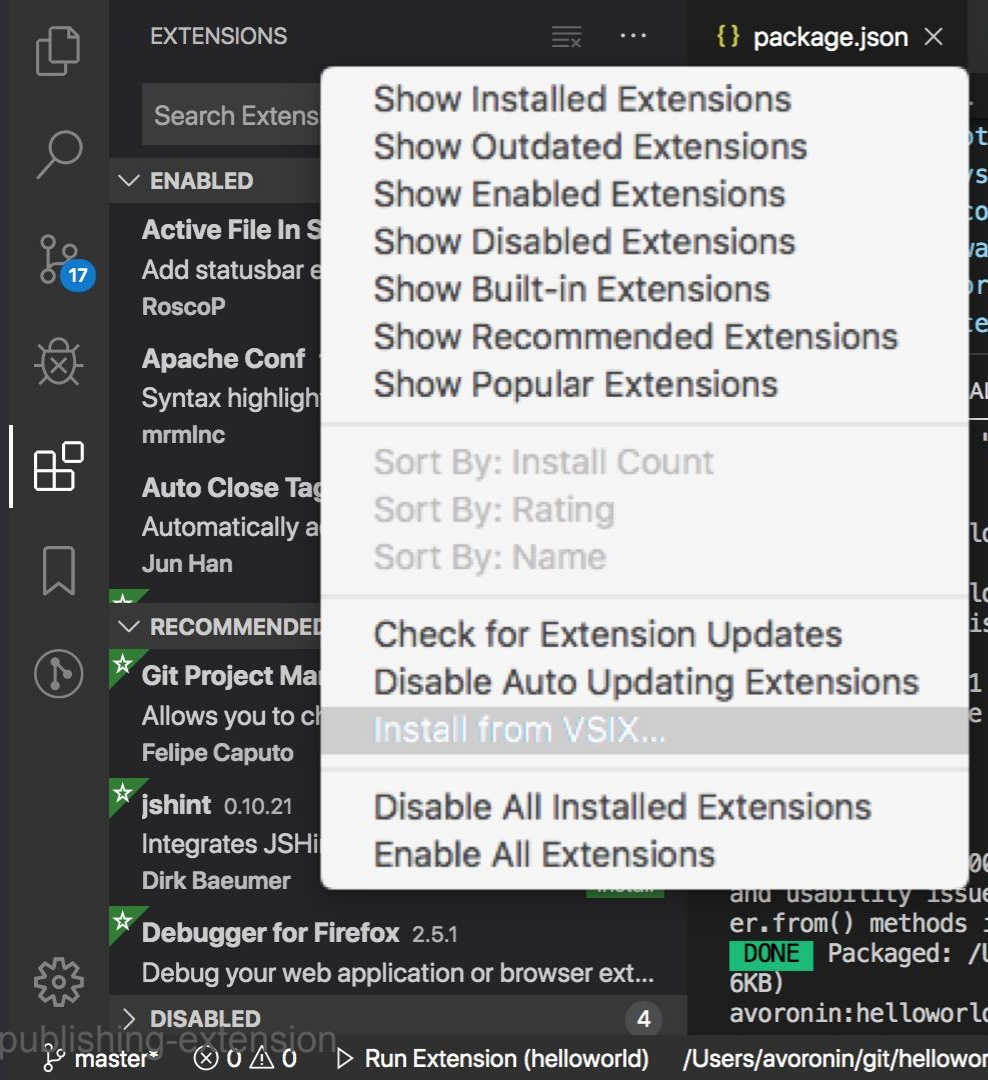
```
Executing prepublish script
      'npm run vscode:prepublish'...

> helloworld@0.0.1 vscode:prepublish
/Users/avoronin/git/cscartcode
> yarn run compile

yarn run v1.3.2
$ tsc -p ./
Done in 3.59s.
 DONE  Packaged: /Users/avoronin/git/cscartcode/
      helloworld-0.0.1.vsix (6 files, 3.6KB)
```

code.visualstudio.com/api/working-with-extensions/publishing-extension

Install

- Command Palette ⇧⌘P
  Install from VSIX

- Extensions:
  [...] → Install from VSIX...

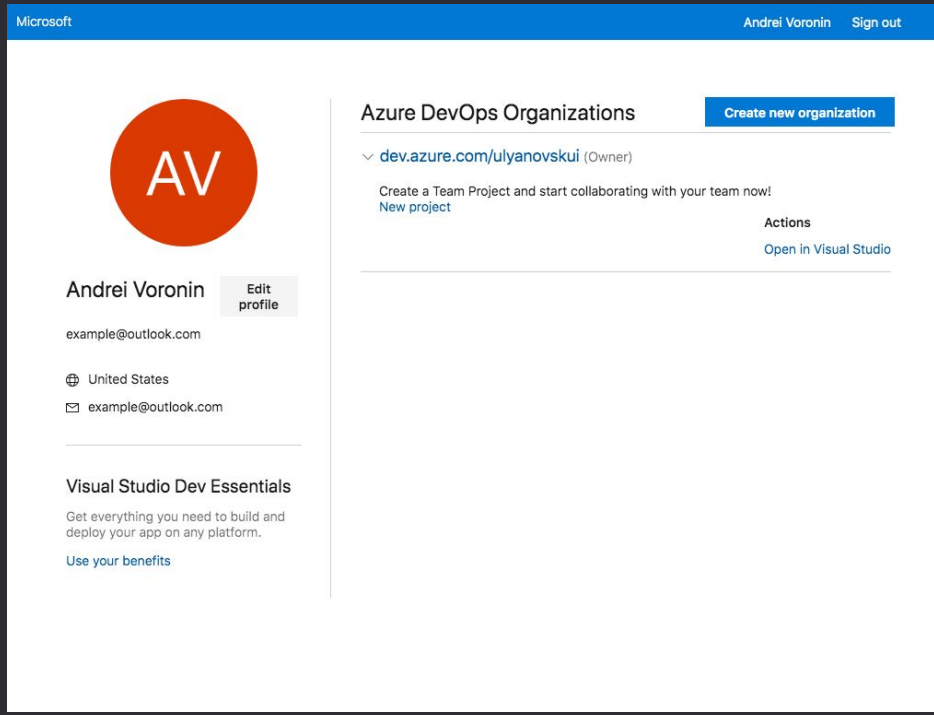


code.visualstudio.com/api/working-with-extensions/publishing-extension

Publishing

Create an organization

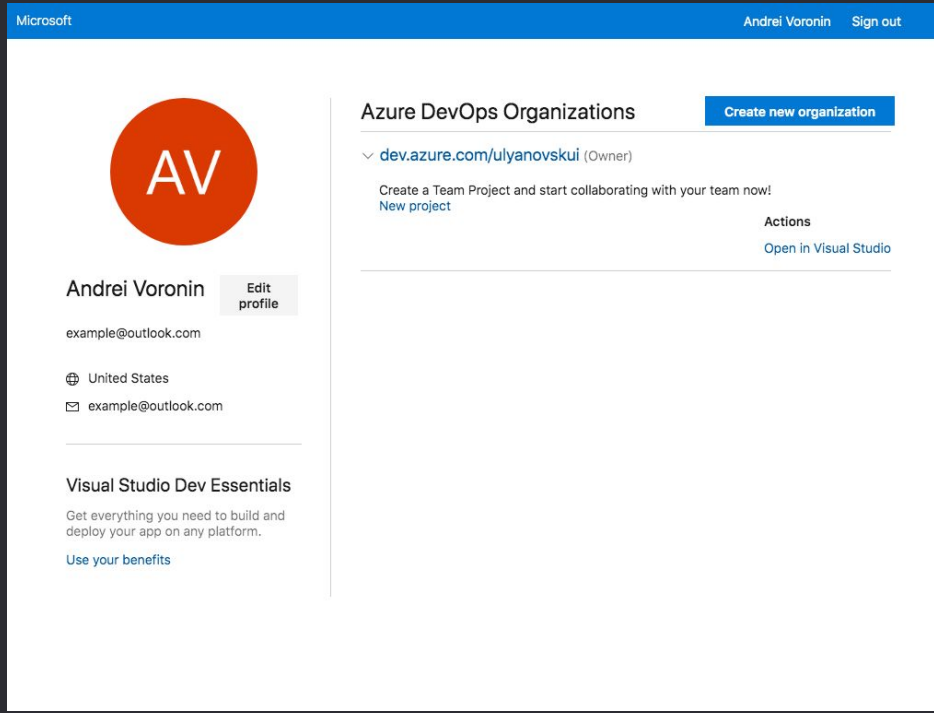
1. Sign in to Azure DevOps
   app.vsaex.visualstudio.com



code.visualstudio.com/api/working-with-extensions/publishing-extension

Create an organization

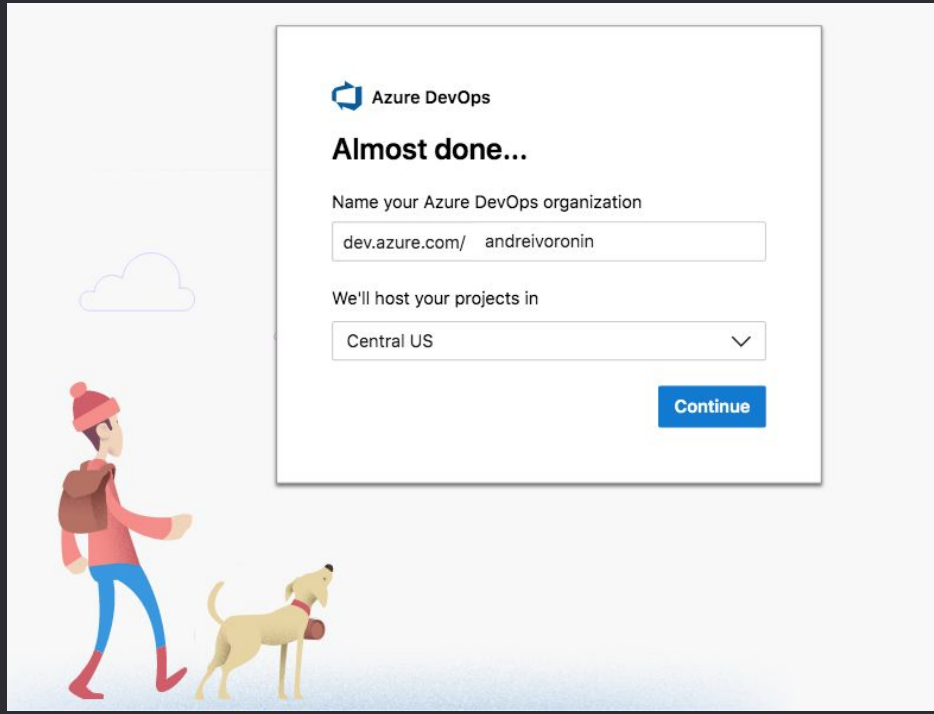
1. Sign in to Azure DevOps
   app.vsaex.visualstudio.com
2. Create new organization



code.visualstudio.com/api/working-with-extensions/publishing-extension

Create an organization

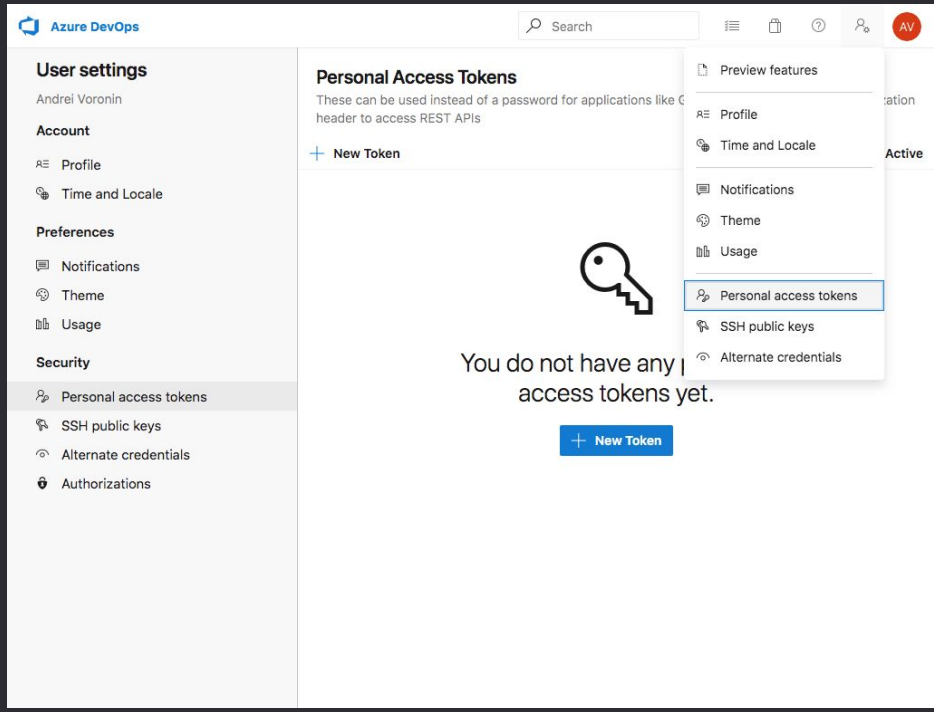
1. Sign in to Azure DevOps app.vsaex.visualstudio.com
2. Create new organization
3. Sign in to dev.azure.com/myorganization



code.visualstudio.com/api/working-with-extensions/publishing-extension

Get a token

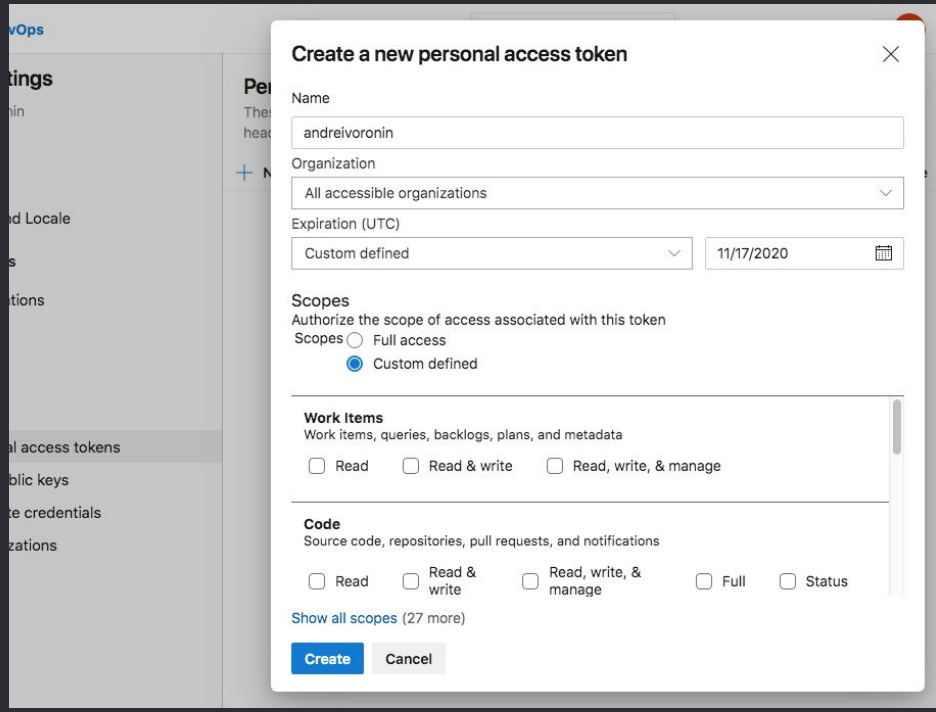
1. User settings → Personal access tokens → New Token



code.visualstudio.com/api/working-with-extensions/publishing-extension

Get a token

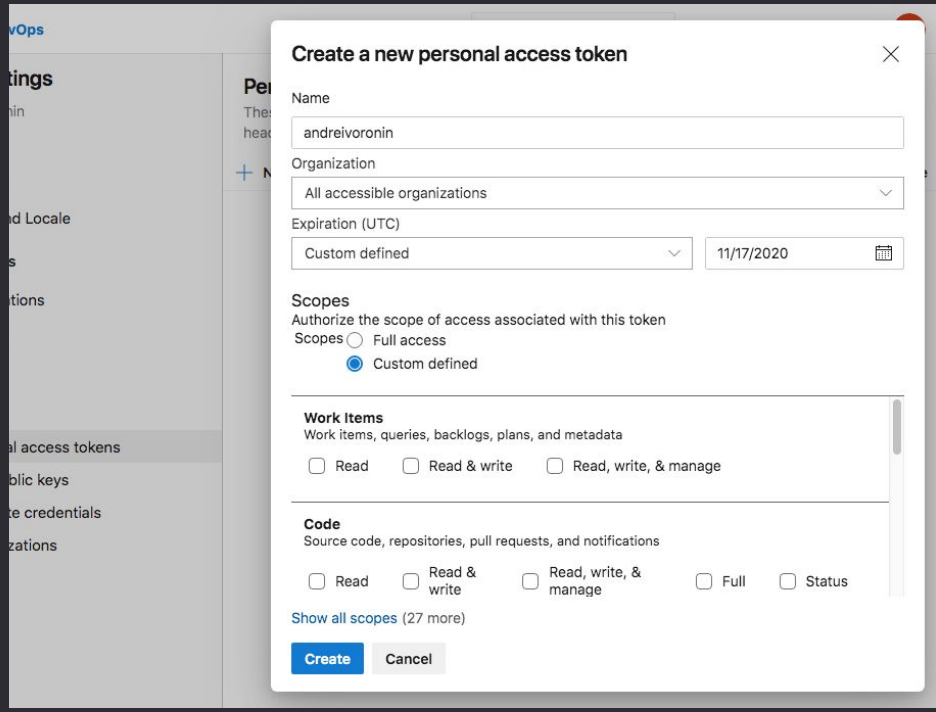
1. User settings → Personal access tokens → New Token
2. Name, Organization



code.visualstudio.com/api/working-with-extensions/publishing-extension

Get a token

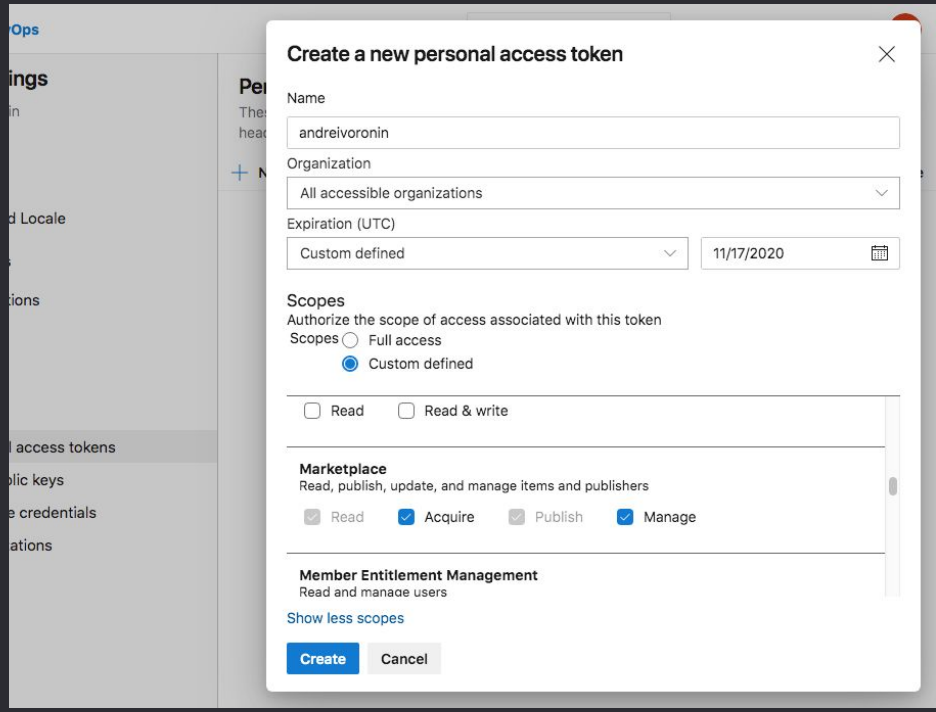
1. User settings → Personal access tokens → New Token
2. Name, Organization
3. Scopes: custom defined



code.visualstudio.com/api/working-with-extensions/publishing-extension

Get a token

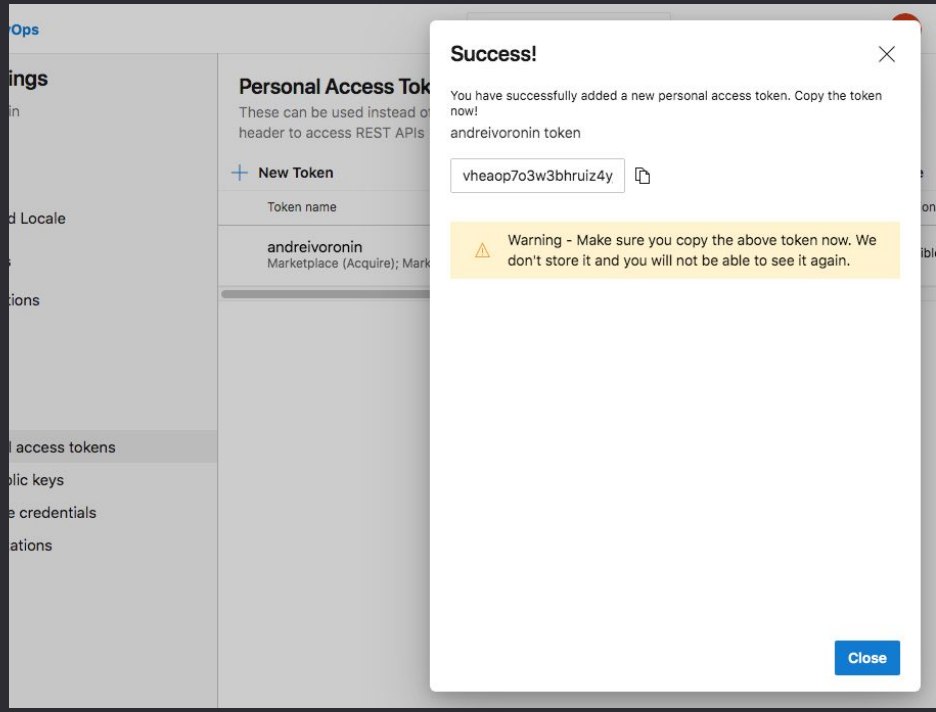
1. User settings → Personal access tokens → New Token
2. Name, Organization
3. Scopes: custom defined
4. Show all scopes → Marketplace: [v] Acquire, [v] Manage



code.visualstudio.com/api/working-with-extensions/publishing-extension

Get a token

1. User settings → Personal access tokens → New Token
2. Name, Organization
3. Scopes: custom defined
4. Show all scopes → Marketplace: [v] Acquire, [v] Manage
5. Create → Copy



code.visualstudio.com/api/working-with-extensions/publishing-extension

Create a publisher

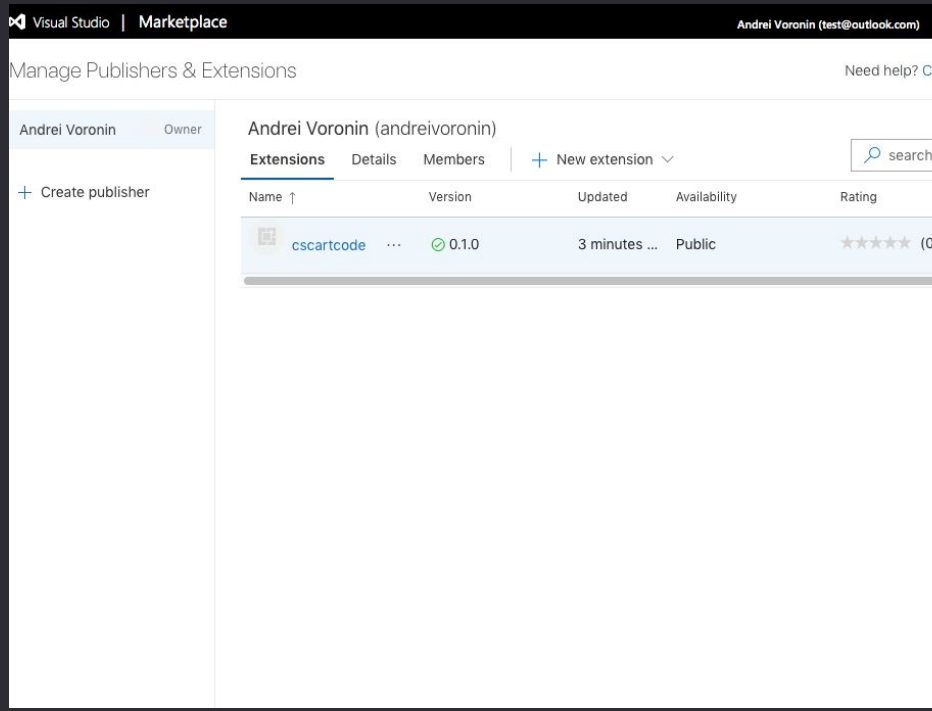
1. vsce create-publisher (publisher name)
2. vsce login (publisher name)
3. Do you want to overwrite its PAT? y

```
vsce create-publisher andreivoronin
Publisher human-friendly name: (testcomp) Andrei Voronin
E-mail: ulyanovskui@example.com
Personal Access Token:
***********************************************

DONE Created publisher 'andreivoronin'.

vsce login andreivoronin
Publisher 'andreivoronin' is already known
Do you want to overwrite its PAT? [y/N] y
Personal Access Token for publisher 'andreivoronin':
***********************************************
```

code.visualstudio.com/api/working-with-extensions/publishing-extension

Publishing

1. vsce publish minor
    a. SemVer
    b. major, minor, or patch

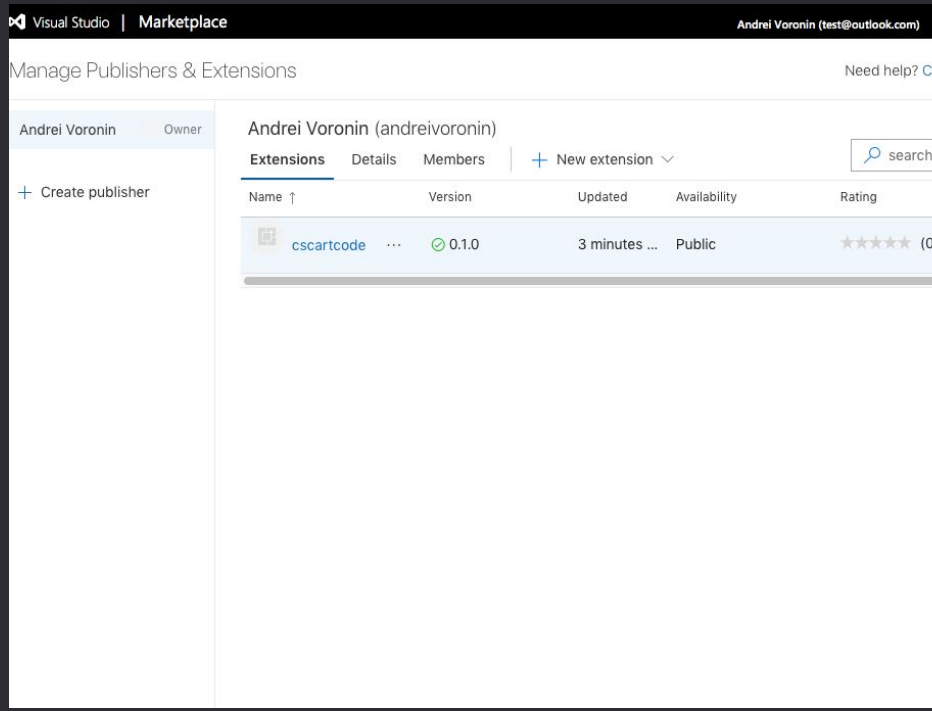


code.visualstudio.com/api/working-with-extensions/publishing-extension

Publishing

1. vsce publish minor
    a. SemVer
    b. major, minor, or patch

2. marketplace.visualstudio.com/manage



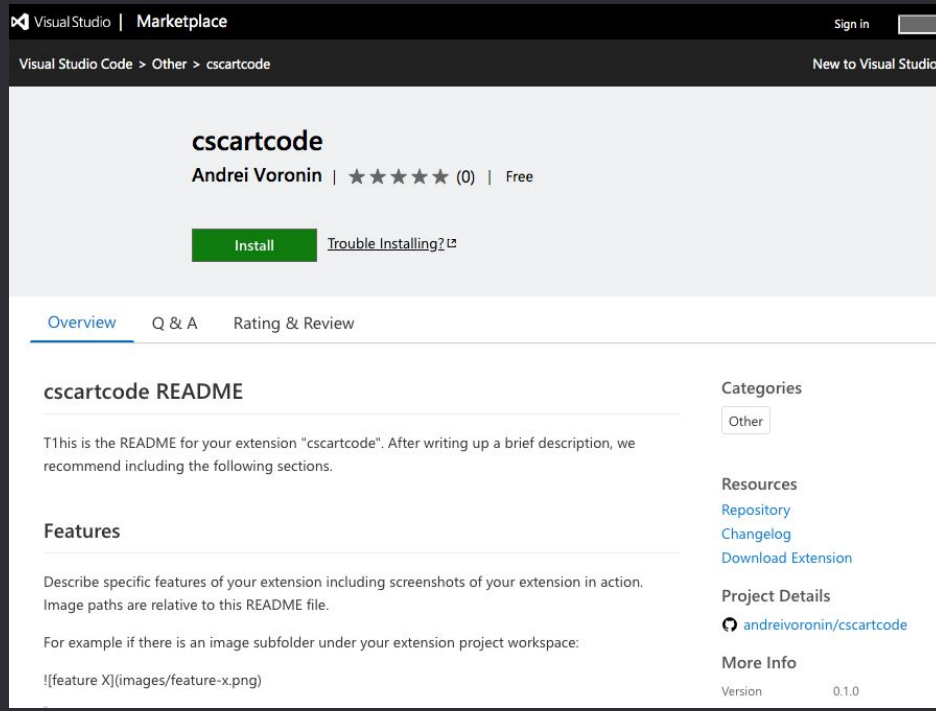
code.visualstudio.com/api/working-with-extensions/publishing-extension

Publishing

1. vsce publish minor
    a. SemVer
    b. major, minor, or patch

2. marketplace.visualstudio.com/manage

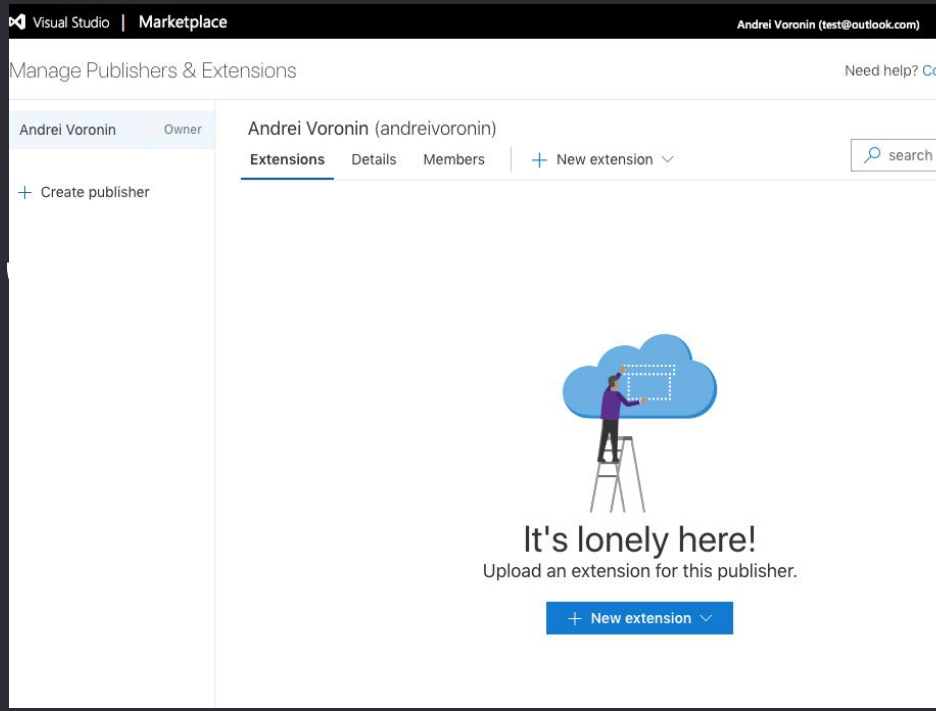
3. marketplace.visualstudio.com/items?itemName=andreivoronin.cscartcode



code.visualstudio.com/api/working-with-extensions/publishing-extension

Unpublishing

1. vsce unpublish publishername.myextension
2. This will FOREVER delete 'publishername.myextension'! Are you sure? [y/N] y



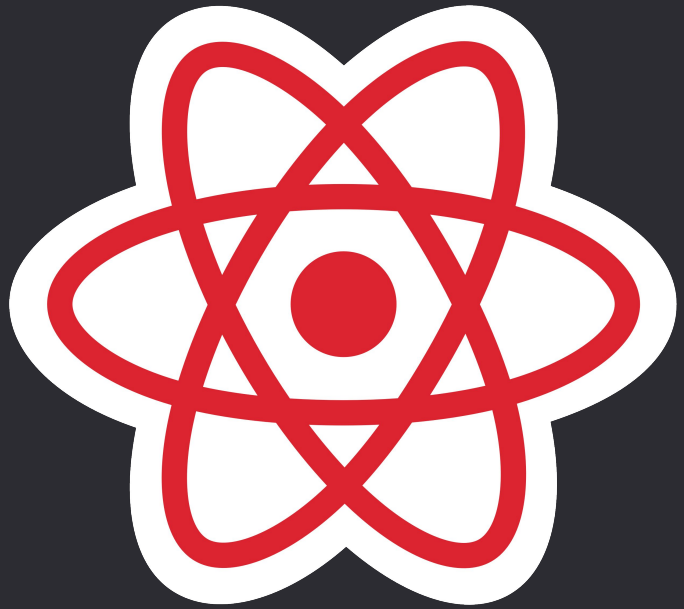
code.visualstudio.com/api/working-with-extensions/publishing-extension

Results

- Run & Debug
- VSCode API
- Packaging & Publishing
- Accelerate development

simtech

Middle React Developer



@UlyanovskUI

Product, Frontend & Design

400 FOLLOWERS **30K /m** TWEETS SHOW **8** YEARS

bit.ly/firstvscode

Andrei Voronin UlyanovskUI

Your First VS Code Extension

Andrei Voronin @UlyanovskUI