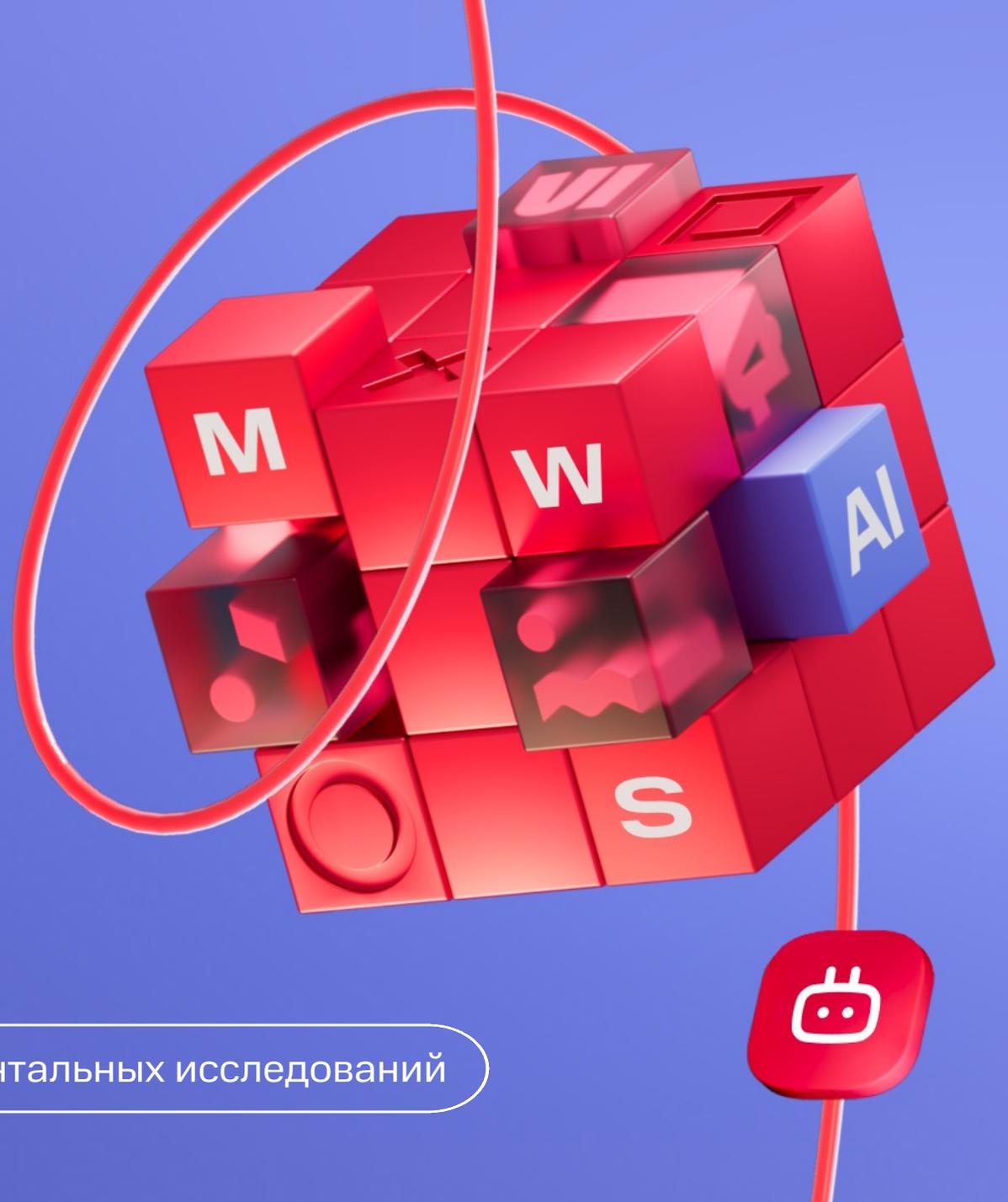


# Как LLM могут помочь безопасности кода?



 Валентин Малых

Руководитель фундаментальных исследований



# MWS AI — один из лидеров в области искусственного интеллекта в России

Более

**200**

экспертов

Реализовано более

**20**

проектов

Статьи на научных конференция уровня А\*



Автоматизировали клиентский сервис МТС с помощью чат-бота

Создали Gen AI помощника для написания кода и чат-бота на основе LLM.

Помогаем прогнозировать доходность инвестиций в телеком-инфраструктуру с помощью ИИ

Научили KION пропускать титры и определять место для рекламы

Более

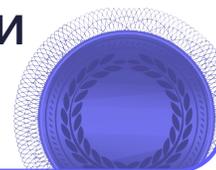
**80 миллионов человек**

пользуются нашими решениями каждый день

Самый мощный в телекоме суперкомпьютер



Участие в конференциях и победы в международных конкурсах по ИИ



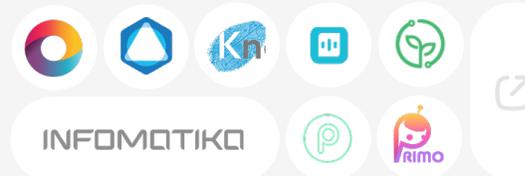
Сообщество

**30+**  
партнеров

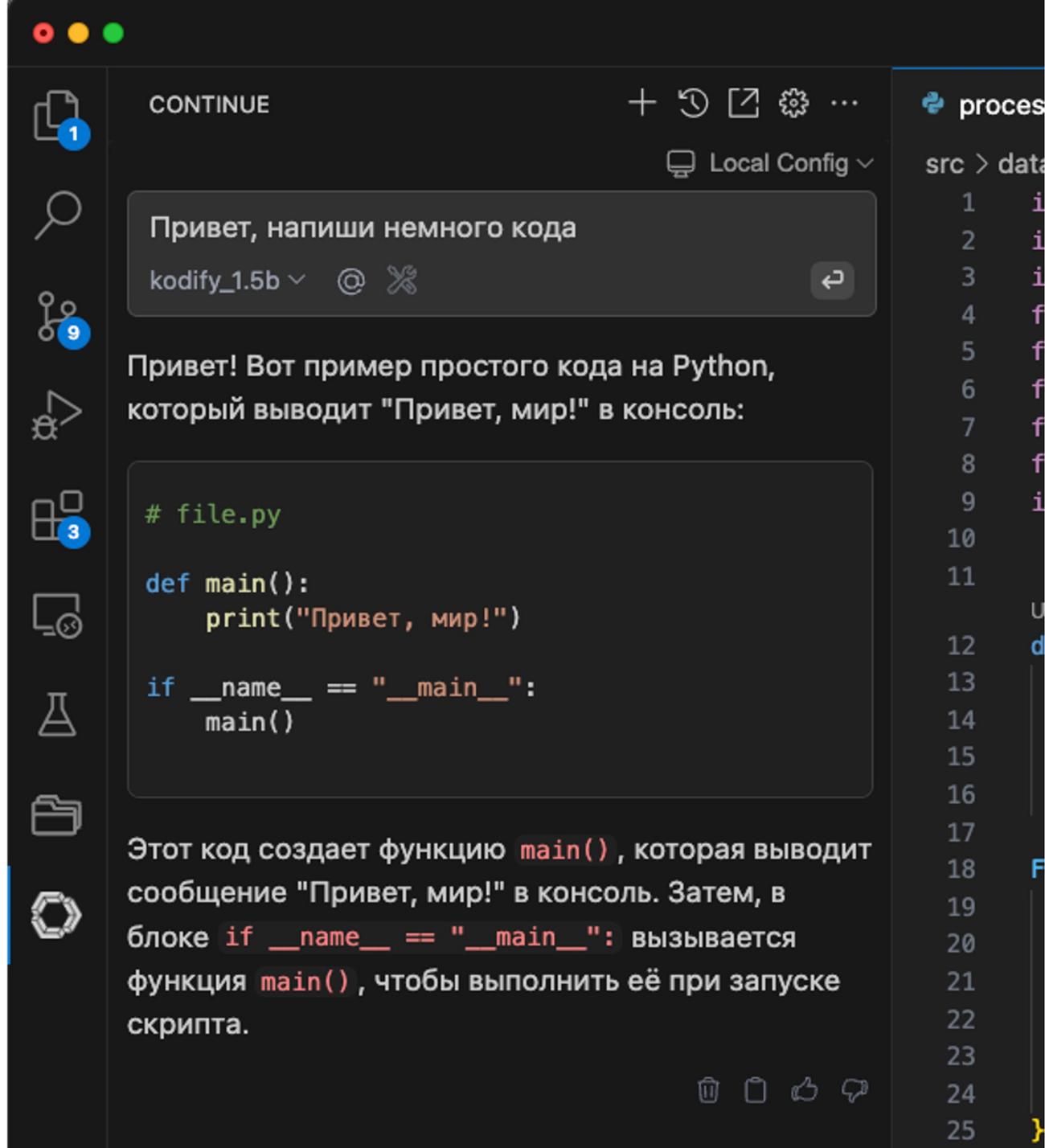
**800**  
стартапов

**8+**

Портфельных компаний



# Что такое помощник программиста?



The screenshot shows a dark-themed code editor with a chat interface. The chat window is titled "CONTINUE" and contains the following text:

Привет, напиши немного кода

Привет! Вот пример простого кода на Python, который выводит "Привет, мир!" в консоль:

```
# file.py

def main():
    print("Привет, мир!")

if __name__ == "__main__":
    main()
```

Этот код создает функцию `main()`, которая выводит сообщение "Привет, мир!" в консоль. Затем, в блоке `if __name__ == "__main__":` вызывается функция `main()`, чтобы выполнить её при запуске скрипта.

The interface includes a sidebar with icons for file explorer, search, and other tools. The chat window has a "Local Config" dropdown and a "CONTINUE" title. The code editor shows a file named "file.py" with the provided Python code. The chat window also has a "Local Config" dropdown and a "CONTINUE" title.

# Автодополнение

```
6 public class CodeCompletion {  
7  
8     private void completion() {  
9         Square square = new Square();  
10        String string = "";  
11        List<String> strings;
```

```
12        square.|
```

m	getId()	String
m	draw()	void
m	equals(Object obj)	boolean
m	hashCode()	int
m	toString()	String
m	getClass()	Class<? extends Square>
m	notify()	void
m	notifyAll()	void
m	wait()	void
m	wait(long timeoutMillis)	void
m	wait(long timeoutMillis, int nanos)	void

# Генерация кода

The screenshot shows the Microsoft Copilot interface within the Power Platform. On the left, the navigation pane is visible with sections for 'POWER PLATFORM', 'AUTH PROFILES', 'ENVIRONMENTS & SOLUTIONS', and 'COPILOT IN POWER PAGES'. The Copilot chat window contains the following text:

Write code in JavaScript to make sure that submitted value for phone number field is in valid format

You may add the following code to your JavaScript file.

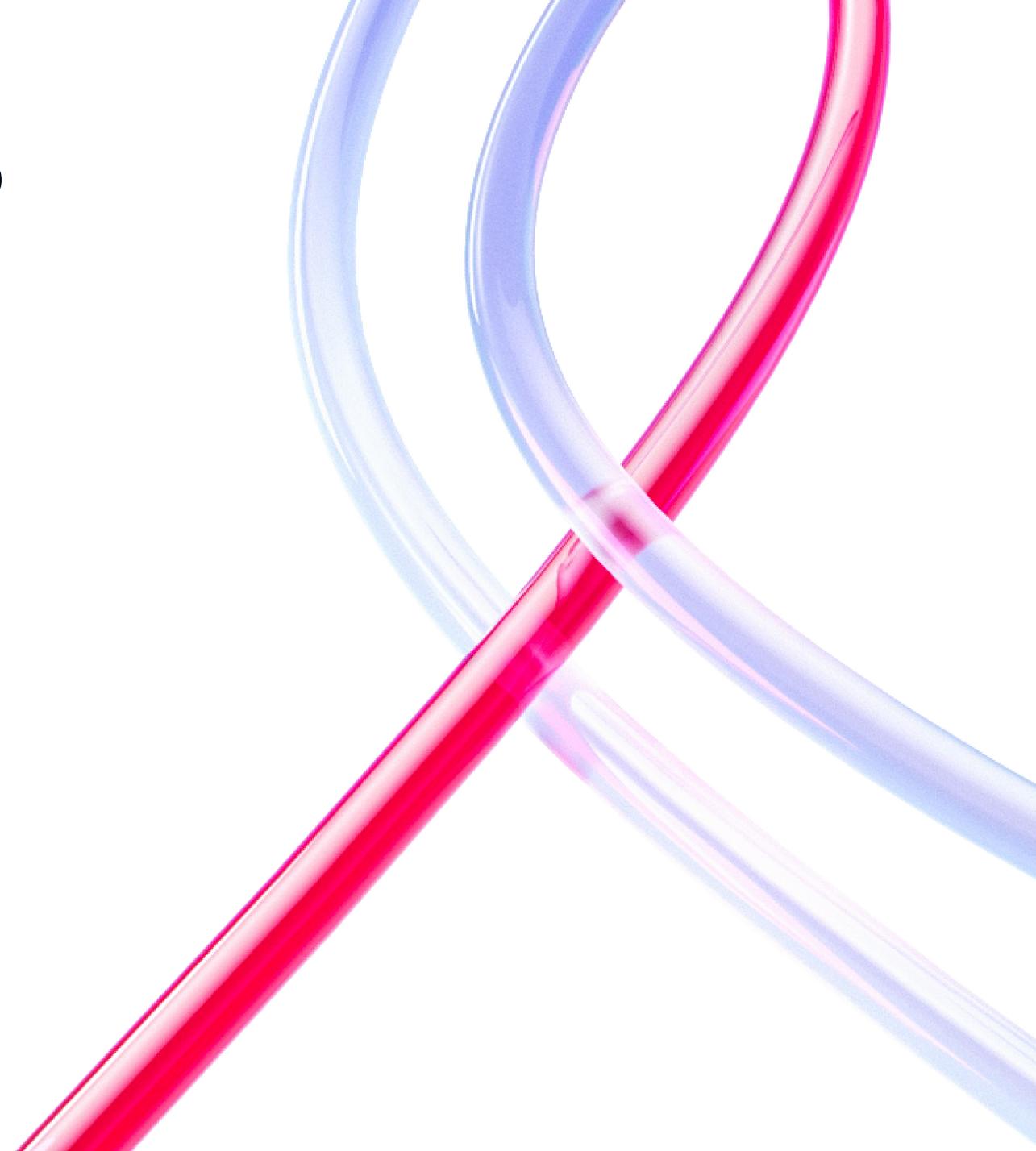
```
function validatePhoneNumber(phoneNumber) {
  const regex = /^\(?\([0-9]{3}\)\)?[-. ]?([0-9]{3})[-
```

The code in the chat window is highlighted with a red box. A red arrow points from this box to the code editor on the right. The code editor shows the following JavaScript code:

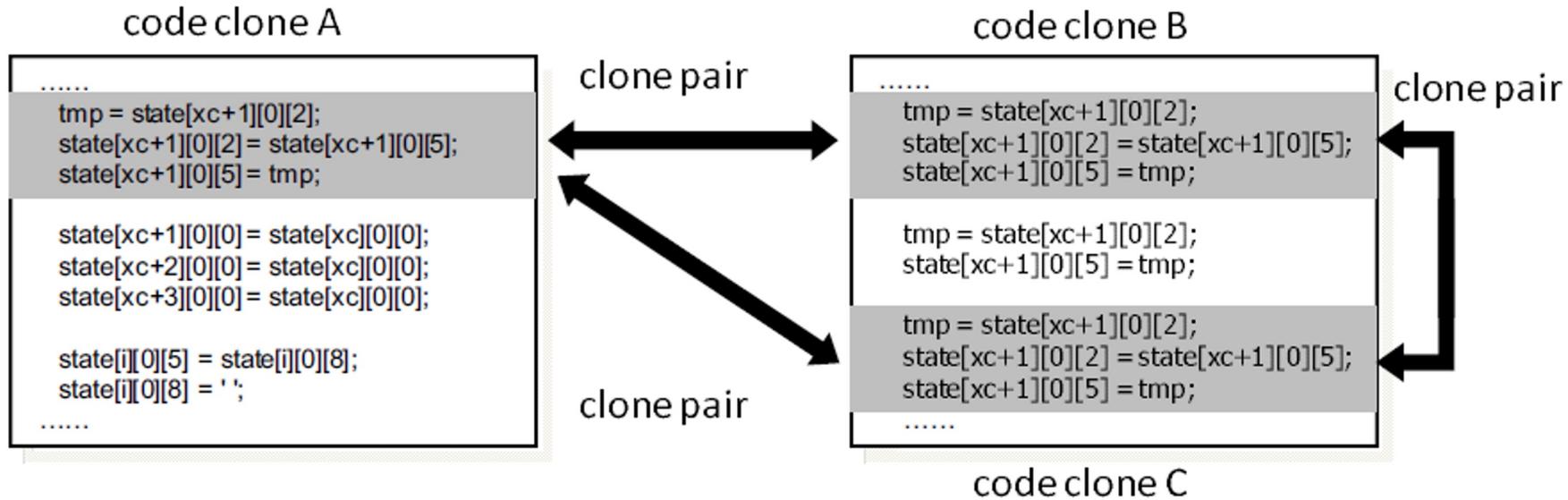
```
function validatePhoneNumber(phoneNumber) {
  const regex = /^\(?\([0-9]{3}\)\)?[-. ]?([0-9]{3})[-
  return regex.test(phoneNumber);
}
```

The code editor also shows the file path: 'basic-forms > copilot-contact-us-ebed > JS Copilot-Contact-Us-ebed.basicform.custom'. The status bar at the bottom indicates 'Ln 5, Col 1', 'Spaces: 4', 'UTF-8', 'CRLF', and 'JavaScript'.

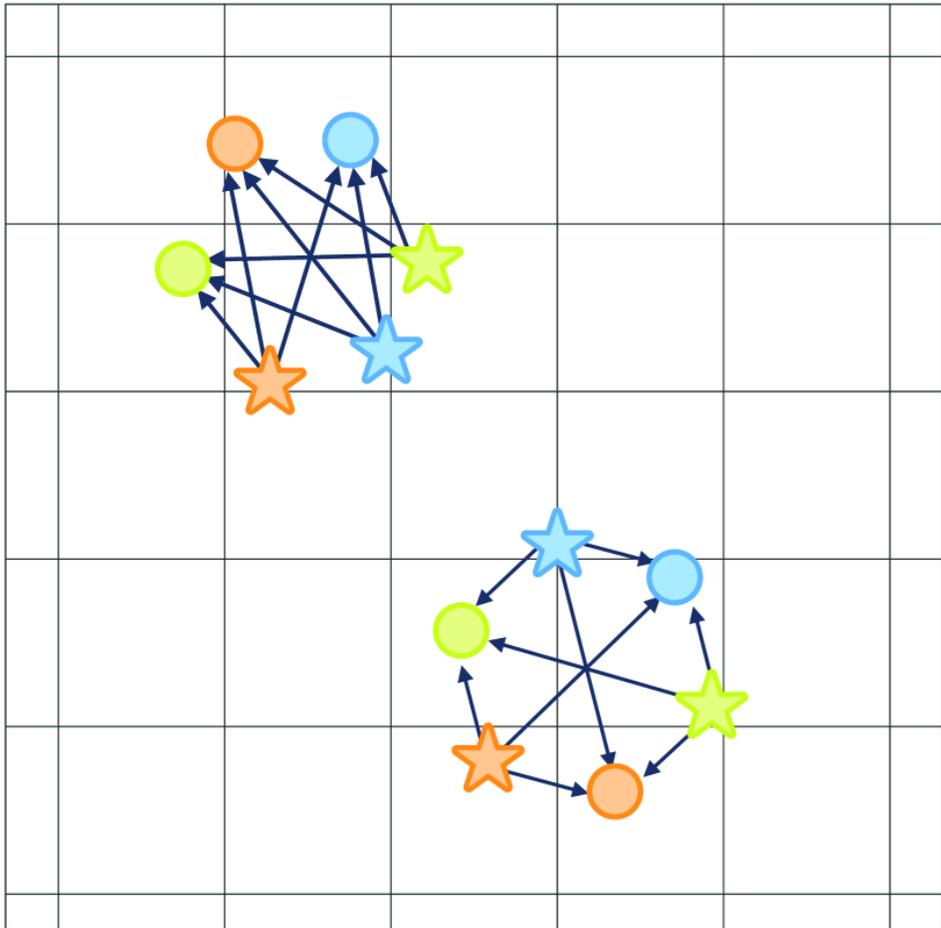
**СЕРВЕРНАЯ ЧАСТЬ**



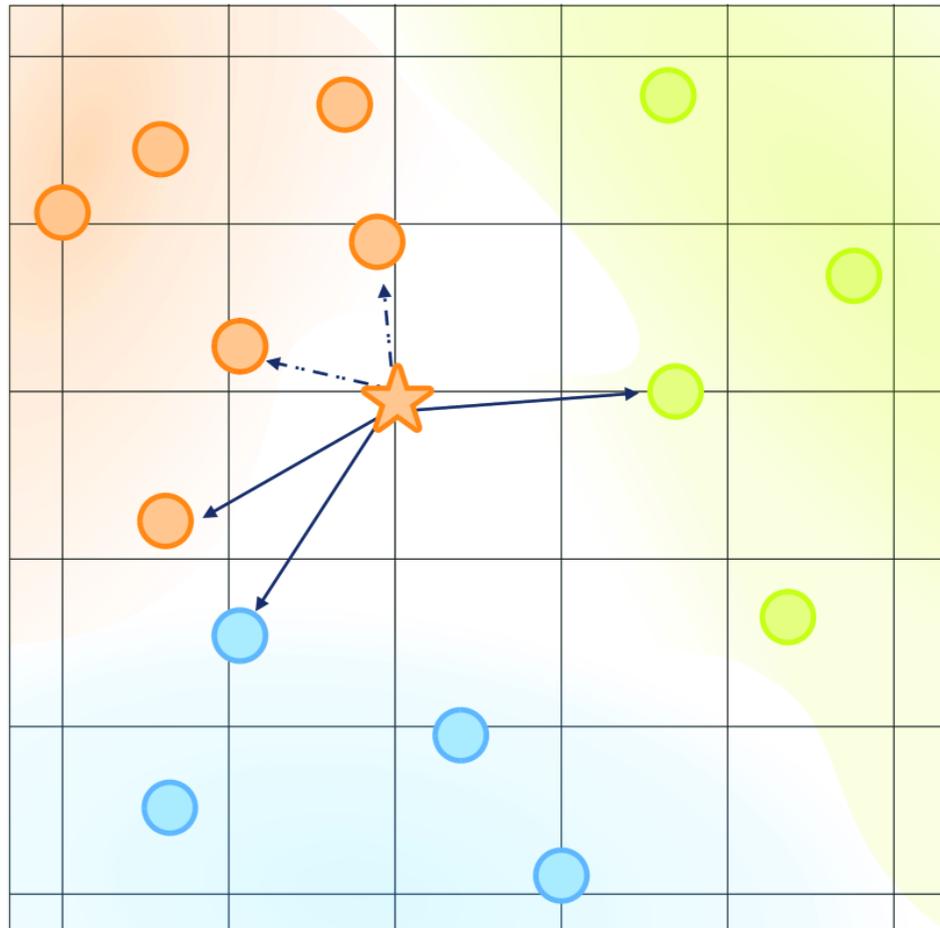
# Clone Detection



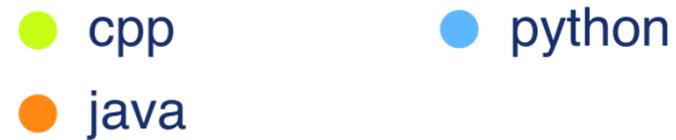
# Выравнивание



Strong Alignment



Weak Alignment

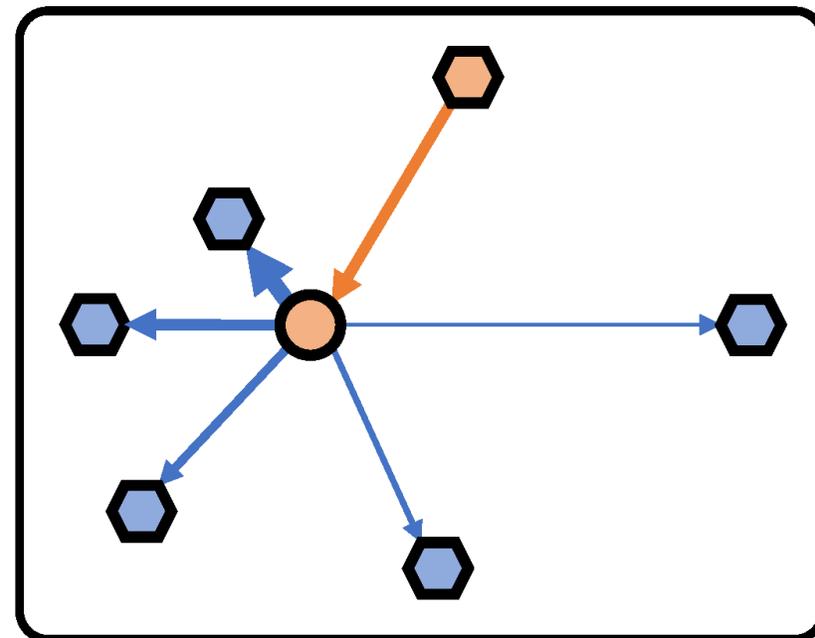
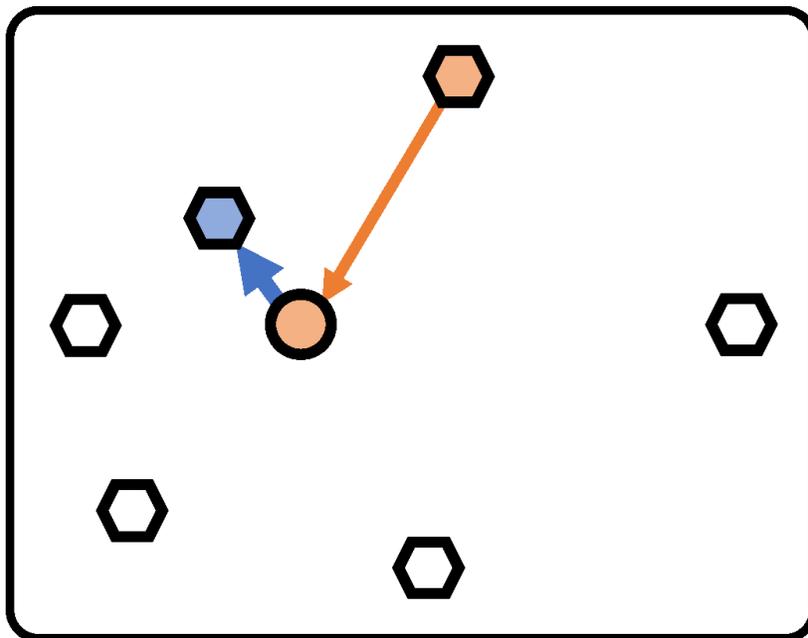
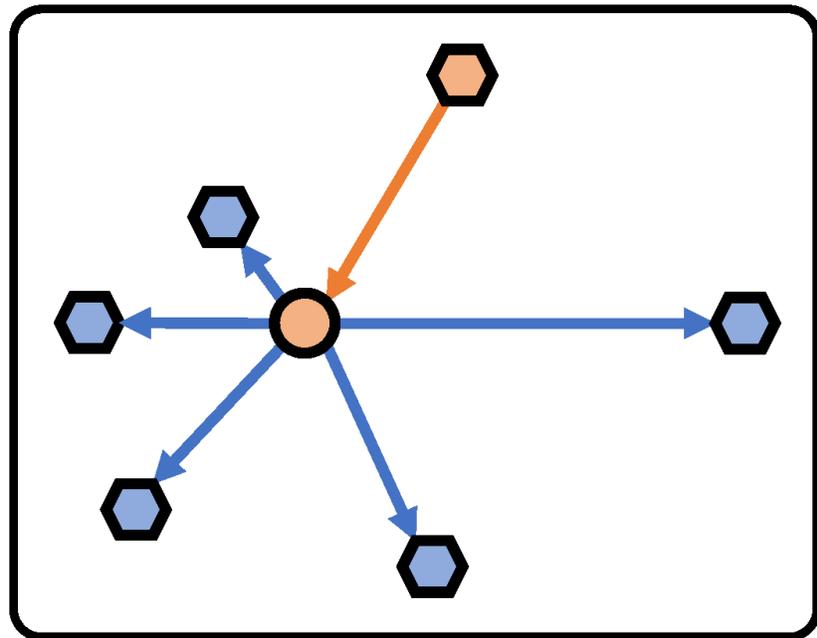


# Как учим

● Anchor      ● Positive

● Negative for training

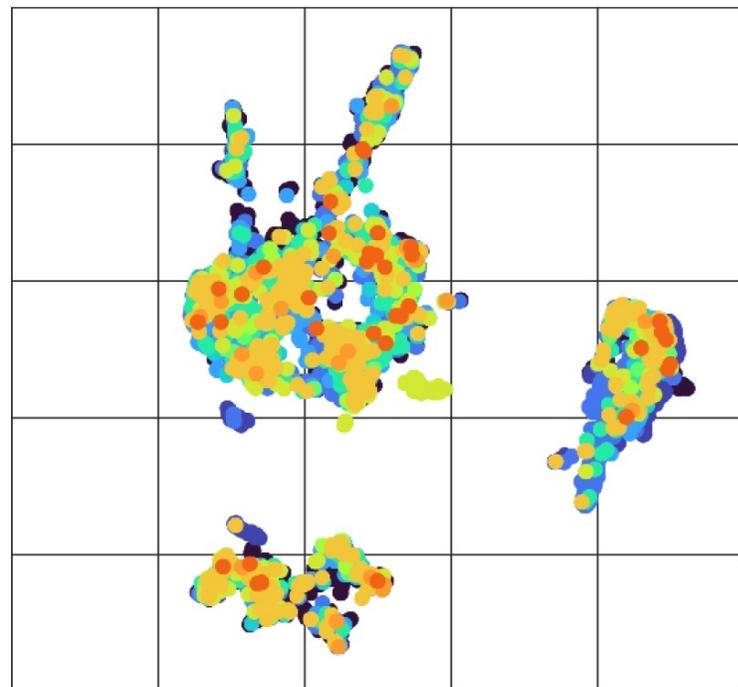
○ Negative not for training



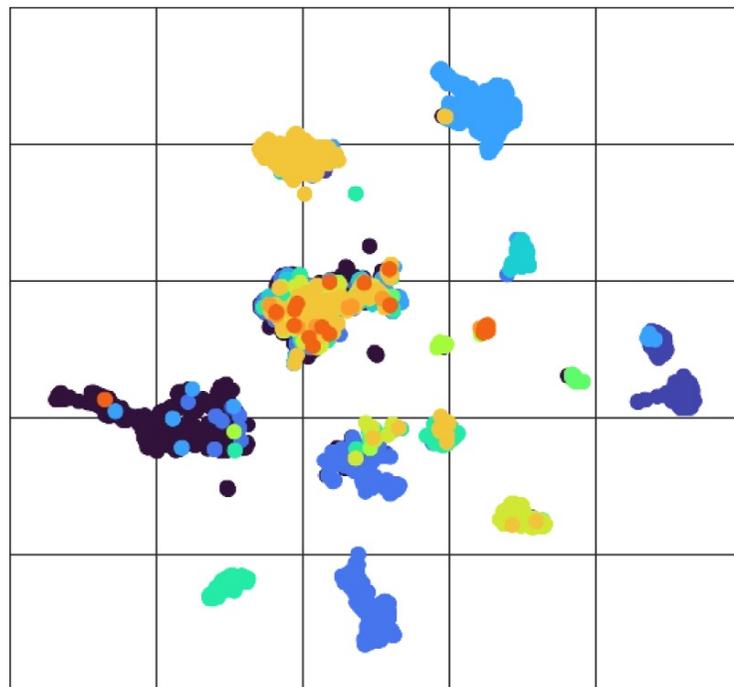
$$\pi_{\theta}(\mathbf{z}|\mathbf{x}) = \frac{e^{s_{\theta}(\mathbf{x}, \mathbf{z})}}{\sum_{\mathbf{z}^- \in \mathcal{B}_{\mathbf{x}, \mathbf{z}}} e^{s_{\theta}(\mathbf{x}, \mathbf{z}^-)} + e^{s_{\theta}(\mathbf{x}, \mathbf{z})}}$$

# Результаты

CodeBERT



CCT



problem    ● D18    ● G656    ● G784    ● D795    ● A795    ● K774  
              ● J795    ● E795    ● D926    ● C953    ● F953    ● F926

	Clone detection (MAP)	Code search (MRR)
<b>Encoder-only</b>		
RoBERTa-base ( <a href="#">Liu et al., 2019</a> )	76.67	18.33
CodeBERT ( <a href="#">Feng et al., 2020</a> )	82.67	27.19
SynCoBERT ( <a href="#">Wang et al., 2021</a> )	88.24	38.10
CodeRoBERTa	—	42.35
GraphCodeBERT ( <a href="#">Guo et al., 2021</a> )	85.16	—
CasCode ( <a href="#">Gotmare et al., 2021</a> )	—	43.98
<a href="#">Villmow et al. (2022)</a>	91.34	—
CCT-LM <sub>enc</sub>	<b>96.73</b>	<b>47.18</b>
<b>Decoder-only</b>		
CodeGen ( <a href="#">Nijkamp et al., 2023</a> )	89.68	—
CodeGPT ( <a href="#">Lu et al., 2021a</a> )	87.96	—
SantaCoder ( <a href="#">Allal et al., 2023</a> )	83.98	—
Phi-1 ( <a href="#">Gunasekar et al., 2023</a> )	92.72	—
CCT-LM <sub>dec</sub>	<b>95.50</b>	—

# CVE Detection

 OpenCVE

acme

MAIN NAVIGATION

 Dashboard

 Projects

 Vulnerabilities

 Vendors & Products

 Weaknesses

 Statistics

SETTINGS

 Organizations

 Tags

## Vulnerabilities

Search

Switch to Basic Search 

vendor:microsoft AND userTag:tocheck AND (cvss31>=9 OR cvss40>=9)

Search

Use the syntax `description:python AND cvss31>=8`. For more details, check the [Advanced Search](#) documentation.

CVE	Vendors	Products
<a href="#">CVE-2024-43639</a> Windows KDC Proxy Remote Code Execution Vulnerability	<b>1</b> <a href="#">Microsoft</a>	<b>6</b> <a href="#">Windows Server 2012, Windows Server 2016, Windows Server 2019 and 3 more</a>
<a href="#">CVE-2024-43498</a> .NET and Visual Studio Remote Code Execution Vulnerability	<b>4</b> <a href="#">Apple, Linux, Microsoft and 1 more</a>	<b>6</b> <a href="#">Macos, Linux Kernel, .net and 3 more</a>
<a href="#">CVE-2024-43602</a> Azure CycleCloud Remote Code Execution Vulnerability	<b>1</b> <a href="#">Microsoft</a>	<b>1</b> <a href="#">Azure Cyclecloud</a>

# Результаты на BigVul

	<b>Precision</b>	<b>Recall</b>	<b>F1</b>
CCT (ours)	0.98	0.98	0.97
VulMixer	0.06	0.62	0.11
MoEVD	0.46	0.42	0.44

**ОБРАТНО НА КЛИЕНТ**



# KODIFY 2

Основан на Qwen 2.5 7B



# KODIFY 2

Основан на Qwen 2.5 7B

Поддерживает 5 языков

- Python
- Java
- JavaScript
- Go
- C#

... еще 90 языков понимает



# KODIFY 2

Основан на Qwen 2.5 7B

Поддерживает 5 языков

Работает с 6 задачами:

- Автодополнение кода
- Генерация кода по запросу
- Генерация юнит-тестов
- Генерация комментариев
- Объяснение кода
- Рефакторинг



# KODIFY 2

Основан на Qwen 2.5 7B

Поддерживает 5 языков

Работает с 6 задачами

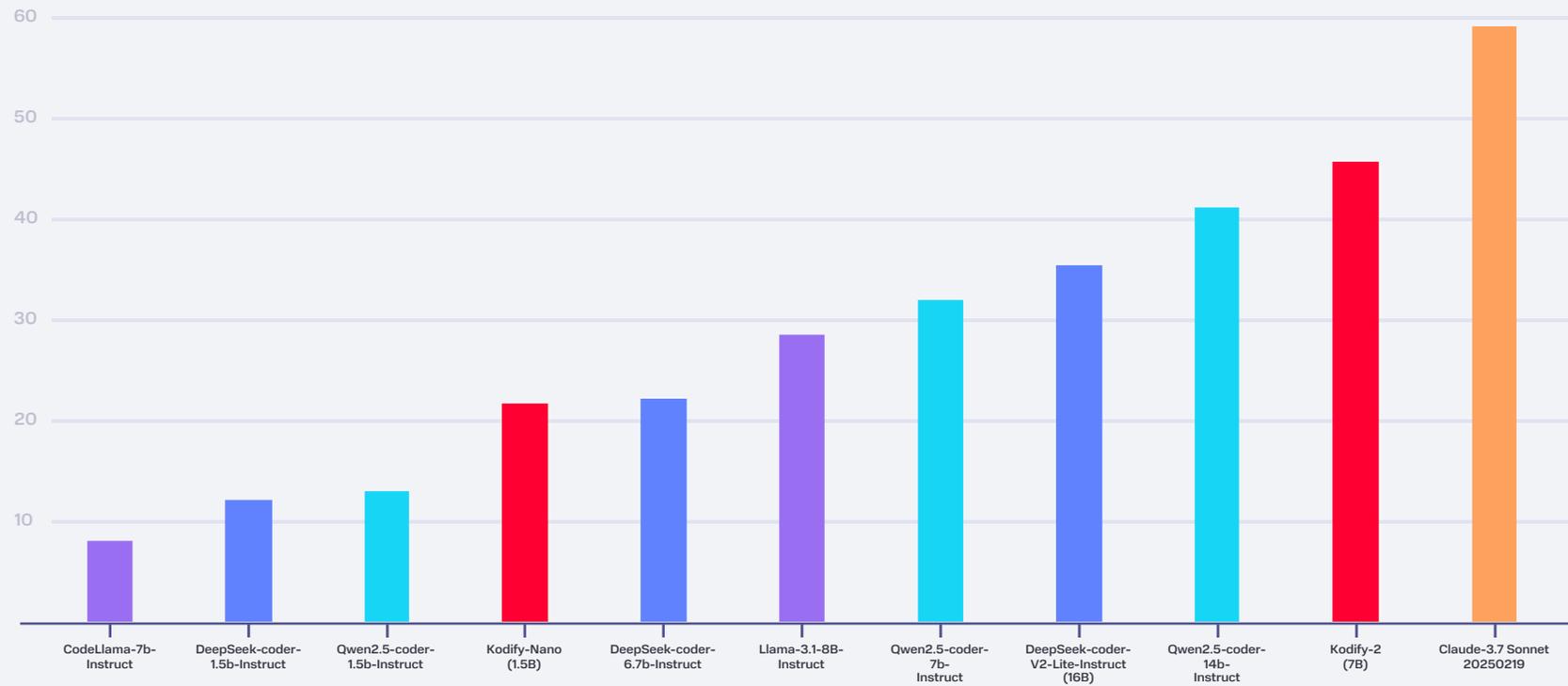


# Оценка

```
36 src/material-examples/dialog-overview/dialog-overview-example.ts View
... @@ -1,24 +1,46 @@
1 -import {Component} from '@angular/core';
2 -import {MdDialog} from '@angular/material';
3
4 /**
5  * @title Dialog Overview
6  */
7 @Component({
8   selector: 'dialog-overview-example',
9 -  templateUrl: 'dialog-overview-example.html',
10 })
11 export class DialogOverviewExample {
12
13   constructor(public dialog: MdDialog) {}
14 -  openDialog() {
15 -    this.dialog.open(DialogOverviewExampleDialog);
16   }
17 -}
18
19 +import {Component, Inject} from '@angular/core';
20 +import {MdDialog, MdDialogRef, MD_DIALOG_DATA} from '@angular/material';
21
22 /**
23  * @title Dialog Overview
24  */
25 @Component({
26   selector: 'dialog-overview-example',
27 +  templateUrl: 'dialog-overview-example.html'
28 })
29 export class DialogOverviewExample {
30 +
31 +  animal: string;
32 +  name: string;
33 +
34   constructor(public dialog: MdDialog) {}
35
36 +  openDialog(): void {
37 +    let dialogRef = this.dialog.open(DialogOverviewExampleDialog, {
38 +      width: '250px',
39 +      data: { name: this.name, animal: this.animal }
40 +    });
41 +
42 +    dialogRef.afterClosed().subscribe(result => {
43 +      console.log('The dialog was closed');
44 +      this.animal = result;
45 +    });
46   }
47 +}
48
```

# KODIFY 2

Сравнение Kodify 2 и других популярных моделей с GitHub Copilot по качеству генерации кода на внутреннем бенчмарке SBS для целевых языков.



Доля «хороших» ответов в SBS-сравнении с GitHub Copilot Chat (GPT4-o в качестве модели-судьи). «Хорошие» ответы – это ответы, которые судья пометил как «Ответ модели X лучше» или «Оба ответа хорошие»

# KODIFY NANO

Основан на Qwen 2.5 1.5B

Поддерживает 5 языков

Работает с 6 задачами



Скачать

# СПАСИБО ЗА ВНИМАНИЕ!

**Валентин Малых**

Руководитель фундаментальных исследований MWS AI

Мой канал



[t.me/valuableai](https://t.me/valuableai)

Канал MWS AI



[t.me/mtsai](https://t.me/mtsai)

# Генерация unit-тестов

Figure 1: Focal Context

```
// Focal Class
public class Calculator {

    // Focal Method
    public float add(float op1, float op2) {
        float result = op1 + op2;
        this.prevScreenValue = this.screenValue;
        this.screenValue = result;
        return result;
    }

    //Constructors
    Calculator();
    Calculator(float value);

    // Public Method Signatures
    public float subtract(float op1, float op2);
    public float multiply(float op1, float op2);
    public float divide(float op1, float op2);
    public void reset();
    public void revertLastOpeartion();
    public float getScreenValue();
    public float getPrevScreenValue();

    // Public Fields
    public float screenValue;
    public float prevScreenValue;
}
```

The diagram illustrates the focal context for the provided code. Brackets on the right side group the code into five focal contexts:

- fm**: Focal Method, covering the `add` method implementation.
- +fc**: Focal Class, covering the entire class definition from `public class` to the closing brace.
- +c**: Constructor, covering the two constructor declarations.
- +m**: Method Signatures, covering the five public method signature declarations.
- +f**: Fields, covering the two public field declarations.

# Генерация комментариев

```
name = "Pankaj" # employee name
id = 100 # employee id

# This function adds the two numbers
def add(x, y):
    return x + y
```

# Объяснение кода

The image shows a code editor window with a Python file named `data_analyzer.py`. The code defines a function `is_integer` to check if a value is an integer and a `main` function that prompts the user for integers and updates maximum and minimum values. An AI assistant chat window is overlaid on the right, titled "Assistant" and "Why We Use While Loops for Indefinite Iteration". The chat shows a user asking "Can you explain why we use a while-loop?" and the assistant providing a detailed explanation and three reasons for using while-loops.

```
3 # Your code here!  
4 # data_analyzer.py  
5  
6 def is_integer(value):  
7     """  
8     Check if the given value is an integer.  
9     """  
10    return value.isdigit() or (value.startswith('-')  
11  
12 def main():  
13    print("Welcome to the Data Analyzer!")  
14    print("Enter integers to build a dataset. Type 's  
15  
16    maximum = None  
17    minimum = None  
18  
19    while True:  
20        user_input = input("Enter a number: ").strip(  
21  
22        if not is_integer(user_input): # Check if in  
23            print("Non-integer input detected. Exitin  
24                break  
25  
26        number = int(user_input)  
27  
28        # Update maximum and minimum values  
29        if maximum is None or number > maximum:  
30            maximum = number  
31        if minimum is None or number < minimum:  
32            minimum = number  
33  
34        if maximum is not None and minimum is not None:  
35            print(f"Maximum: {maximum}")  
36            print(f"Minimum: {minimum}")  
37        else:  
38            print("No valid integers were entered.")  
39  
40 if __name__ == "__main__":  
41     main()  
42
```

Assistant

### Why We Use While Loops for Indefinite Iteration

Evie Christopher

Can you explain why we use a while-loop? 18:22

Certainly! I'd be happy to explain why we use while-loops in Python and programming in general.

A while-loop is used when you want to repeat a block of code as long as a certain condition is true. Here are the main reasons we use while-loops:

1. Indefinite iteration: When you don't know in advance how many times you need to repeat a process, a while-loop is ideal. It continues until a specific condition is no longer met.
2. Condition-based execution: While-loops are perfect for situations where you need to keep doing something until a particular state or condition is achieved.
3. User input validation: They're often used to repeatedly prompt for user input until valid data is provided.

Type your message here

Run Hand In

# Рефакторинг

```
1 //Bad practice
2 for(let i = 0; i < 50; i++){
3     //do something
4 }
5 //Good practice
6 let NUMBER_OF_STUDENTS= 50
7 for(let i = 0; i < NUMBER_OF_STUDENTS; i++){
8     //do something
9 }
```



# KODIFY 2

Основан на Qwen 2.5 7B

Поддерживает 5 языков

Работает с 6 задачами

Что-то еще?



# Много чего еще

Category	Task	Dataset Name	Language	Train/Dev/Test Size	Baselines	Task definition
Code-Code	Clone Detection	BigCloneBench	Java	900K/416K/416K	CodeBERT	Predict semantic equivalence for a pair of codes.
		POJ-104	C/C++	32K/8K/12K		Retrieve semantically similar codes.
	Defect Detection	Devign	C	21k/2.7k/2.7k		Identify whether a function is vulnerable.
	Cloze Test	CT-all	Python, Java, PHP, JavaScript, Ruby, Go	-/-/176k		Tokens to be predicted come from the entire vocab.
		CT-max/min	Python, Java, PHP, JavaScript, Ruby, Go	-/-/2.6k		Tokens to be predicted come from {max, min}.
	Code Completion	PY150	Python	100k/5k/50k	CodeGPT	Predict following tokens given contexts of codes.
		GitHub Java Corpus	Java	13k/7k/8k		
	Code Repair	Bugs2Fix	Java	98K/12K/12K	Encoder-Decoder	Automatically refine codes by fixing bugs.
Code Translation	CodeTrans	Java-C#	10K/0.5K/1K	Translate the codes from one programming language to another programming language.		
Text-Code	NL Code Search	CodeSearchNet, AdvTest	Python	251K/9.6K/19K	CodeBERT	Given a natural language query as input, find semantically similar codes.
		CodeSearchNet, WebQueryTest	Python	251K/9.6K/1k		Given a pair of natural language and code, predict whether they are relevant or not.
	Text-to-Code Generation	CONCODE	Java	100K/2K/2K	CodeGPT	Given a natural language docstring/comment as input, generate a code.
Code-Text	Code Summarization	CodeSearchNet	Python, Java, PHP, JavaScript, Ruby, Go	908K/45K/53K	Encoder-Decoder	Given a code, generate its natural language docstring/comment.
Text-Text	Documentation Translation	Microsoft Docs	English-Latvian/Danish/Norwegian/Chinese	156K/4K/4K		Translate code documentation between human languages (e.g. En-Zh), intended to test low-resource multi-lingual translation.

# Code Translation

**Input:** A Java method

```
public void print(boolean b)
{
    print(String.valueOf(b));
}
```



**Output:** Its C# version

```
public void print(bool b)
{
    print(b.ToString());
}
```

# Оценка

- **Side-by-Side**
  - ручной
  - автоматический

# Side-by-Side

- **сравниваемся с GitHub CoPilot**



# Side-by-Side

- сравниваемся с GitHub CoPilot
- **2500 пользовательских запросов**



# Side-by-Side

- сравниваемся с GitHub CoPilot
- 2500 пользовательских запросов
- **судья GPT4o**



# Side-by-Side

- сравниваемся с GitHub CoPilot
- 2500 пользовательских запросов
- судья GPT4o
- **промт:**

«Твоя задача – выступить в качестве объективного и строгого судьи, оценивая ответы двух ИИ-помощников по коду на запрос пользователя (программиста). Выяви и исправь любые ошибки в коде. Выбери помощника, который лучше следует инструкциям пользователя и отвечает на вопрос более качественно. Твоя оценка должна учитывать такие факторы, как полезность, релевантность, точность, глубина, отсутствие ошибок в коде, стиль кода.

После предоставления пояснения выведи два значения: 0 или 1, указывающие баллы для помощника А и Б, соответственно, по правилу: если один ответ намного лучше другого, то лучший ответ – 1, а худший – 0, если ответы одинаково хороши – оба 1, если ответы одинаково плохи – оба 0.

Выведи свой окончательный вердикт, строго следуя этому формату:  
[[{{Оценка помощника А}} {{Оценка помощника Б}}]]

[Начало запроса пользователя]

<<Сюда вставляется текст запроса пользователя>>

[Конец запроса пользователя]

[Начало ответа ассистента А]

<<Сюда вставляется текст ответа ассистента А>>

[Конец ответа ассистента А]

[Начало ответа ассистента Б]

<<Сюда вставляется текст ответа ассистента Б>>

[Конец ответа ассистента Б]>>

# Side-by-Side

- сравниваемся с GitHub CoPilot
- 2500 пользовательских запросов
- судья GPT4o
- промт
- **формула сравнения**

$$M = \frac{(\text{win\_A} + \text{win\_tie})}{(\text{win\_A} + \text{win\_tie} * 2 + \text{win\_B})},$$

где:

win\_A – число сравнений, где победила модель A (соответствует вердикту судьи [[1 0]])

win\_B – число сравнений, где победила модель B (соответствует вердикту судьи [[0 1]])

win\_tie – число сравнений, где ответы оказались одинаково хорошими (соответствует вердикту судьи [[1 1]]).