

Министерство Образования Российской Федерации
Пермский Государственный Национальный Исследовательский университет
Государственного и муниципального управления

Отчет по Лабораторной работе №6

Выполнили студенты ИТ 7-8, 1 курс
Чугарина Милана Денисовна
Тагиров Адис
Бормотова Ксения
Руководитель Иванова Е.А.

г.Пермь, 2025

Крестики-нолики - Проектная работа по Git

Работа на GitHub: <https://github.com/milanaaaaaaaa/tic-tac-toe>

Участники команды

Роль	Имя
Разработчик 1	[Милана Чугарина]
Разработчик 2	[Тагиров Адис]
Разработчик 2	[Ксения Бормотова]

Роли:

Тагиров Адис: код игры

Чугарина Милана: работа с терминалом, выгрузка на GitHub

Бормотова Ксения: отчёт, презентация

Описание проекта

Консольная игра "Крестики-нолики" для двух игроков. Игроки поочередно делают ходы на поле 3x3, пытаясь составить линию из трёх своих символов (крестиков или ноликов).

Создание проекта и написание кода на Python

1. Создание структуры проекта

```
milana@milana:~$ mkdir tic-tac-toe  
milana@milana:~$ cd tic-tac-toe
```

2. Создание файлов

```
milana@milana:~/tic-tac-toe$ touch main.py game_logic.py display.py .gitignore README.md
```

3. Написание кода

main.py

```
1   from game_logic import Game
2   from display import print_board
3
4   def main():
5       print("🎮 Добро пожаловать в игру 'Крестики-нолики'!")
6       print("Игрок 1: X | Игрок 2: 0")
7       print()
8
9       game = Game()
10      current_player = "X"
11
12      while True:
13          print_board(game.board)
14          try:
15              move = int(input(f"Ход игрока {current_player}. Введите номер клетки (1-9): ")) - 1
16              if move < 0 or move > 8:
17                  print("🔴 Неверный номер клетки. Попробуйте снова.")
18                  continue
19              if game.make_move(move, current_player):
20                  if game.check_winner(current_player):
21                      print_board(game.board)
22                      print(f"🎉 Игрок {current_player} победил!")
23                      break
24                  elif game.is_board_full():
25                      print_board(game.board)
26                      print("🤝 Ничья!")
27                      break
28                  current_player = "0" if current_player == "X" else "X"
29              else:
30                  print("🔴 Клетка занята. Попробуйте снова.")
31          except ValueError:
32              print("🔴 Введите число от 1 до 9.")
33
34  if __name__ == "__main__":
35      main()
```

game_logic.py

```

1  v  class Game:
2      def __init__(self):
3          self.board = [" " for _ in range(9)] # Поле 3x3 в виде списка
4
5  v  def make_move(self, position, player):
6      """Поставить символ игрока на позицию, если она свободна."""
7      if self.board[position] == " ":
8          self.board[position] = player
9          return True
10     return False
11
12 v  def check_winner(self, player):
13     """Проверить, выиграл ли игрок."""
14     win_conditions = [
15         [0, 1, 2], [3, 4, 5], [6, 7, 8], # строки
16         [0, 3, 6], [1, 4, 7], [2, 5, 8], # столбцы
17         [0, 4, 8], [2, 4, 6]           # диагонали
18     ]
19     for condition in win_conditions:
20         if all(self.board[i] == player for i in condition):
21             return True
22     return False
23
24     def is_board_full(self):
25         """Проверить, заполнено ли поле."""
26         return " " not in self.board

```

display.py

```

1  v  def print_board(board):
2      """Вывести игровое поле в консоль."""
3      print("-----")
4      for i in range(0, 9, 3):
5          print(f" | {board[i]} | {board[i+1]} | {board[i+2]} |")
6          print("-----")

```

```

milana@milana:~/tic-tac-toe$ nano main.py
milana@milana:~/tic-tac-toe$ nano game_logic.py
milana@milana:~/tic-tac-toe$ nano display.py
milana@milana:~/tic-tac-toe$ nano main.py
milana@milana:~/tic-tac-toe$ nano .gitignore
milana@milana:~/tic-tac-toe$ nano README.md
milana@milana:~/tic-tac-toe$ python main.py
Command 'python' not found, did you mean:
    command 'python3' from deb python3
    command 'python' from deb python-is-python3
milana@milana:~/tic-tac-toe$ python3 main.py

```

Настройка Git и инициализация репозитория

1. Инициализация Git

```
milana@milana:~/tic-tac-toe$ git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
```

2. Настройка пользователя Git

```
git config --global user.email "mila.chugarina22@icloud.com"
git config --global user.name "Milana"
```

3. Создание .gitignore

```
GNU nano 7.2
.gitignore
__pycache__/
*.py[cod]
*$py.class
.idea/
.vscode/
venv/    []
ENV/
*.log
logs/
```

4. Добавление файлов и первый коммит

```
milana@milana:~/tic-tac-toe$ git add .
git commit -m "feat: initial commit with tic-tac-toe game"
```

Создание репозитория на GitHub

1. Перейти на github.com/new
2. Назвать репозиторий tic-tac-toe

Привязка локального репозитория к GitHub

1. Привязка

```
milana@milana:~/tic-tac-toe$ git remote add origin https://github.com/milanaaaaaaaa/tic-tac-toe.git
git remote add origin https://github.com/milanaaaaaaaa/tic-tac-toe.git
```

2. Переименование ветки и выгрузка

```
milana@milana:~/tic-tac-toe$ git branch -M main
git push -u origin main
git branch -M main
git push -u origin main
```

Функционал

1. Ввод хода через цифры 1–9.
2. Проверка победы по строкам, столбцам и диагоналям.

3. Проверка ничьей.
4. Обработка ошибок ввода.

Проверка истории комитов

```
git log --oneline
```

```
milana@milana:~/tic-tac-toe$ git log --oneline
5ea03b3 (HEAD -> main) feat: initial commit with tic-tac-toe game
```

Запуск игры

```
python3 main.py
```

```
milana@milana:~/tic-tac-toe$ python3 main.py
🎮 Добро пожаловать в игру 'Крестики-нолики'!
Игрок 1: X | Игрок 2: O

-----
|   |   |   |
-----
|   |   |   |
-----
|   |   |   |           []
-----
Ход игрока X. Введите номер клетки (1-9): 1
-----
| X |   |   |
-----
|   |   |   |
-----
|   |   |   |
-----
Ход игрока O. Введите номер клетки (1-9): 2
-----
| X | O |   |
-----
|   |   |   |
-----
```