מסלול DEVOPS

שם המרצה: אלכס קומנוב

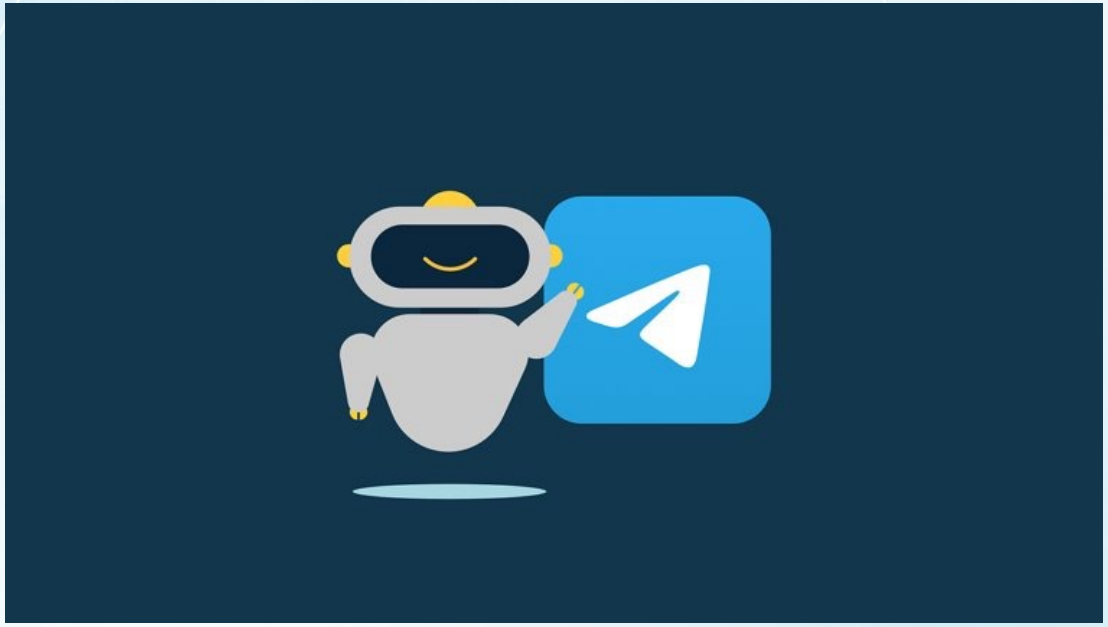
INT LIVE
המרכז הבינלאומי
ללימודי הייטק וחדשנות

PYTHON COURSE

Python Challenge

Challenge

You are going to build a **Telegram bot.**



Challenge

First of all, create a new Python project. Also, create a new GitHub repository and connect your Python project to the repository.

Challenge

Make sure you have or install the required packages (use **pip install** or **pip3 install**):

- aiogram
- python-dotenv

Challenge

In the project root directory create a **.gitignore** file and fill it with the following:

.gitignore

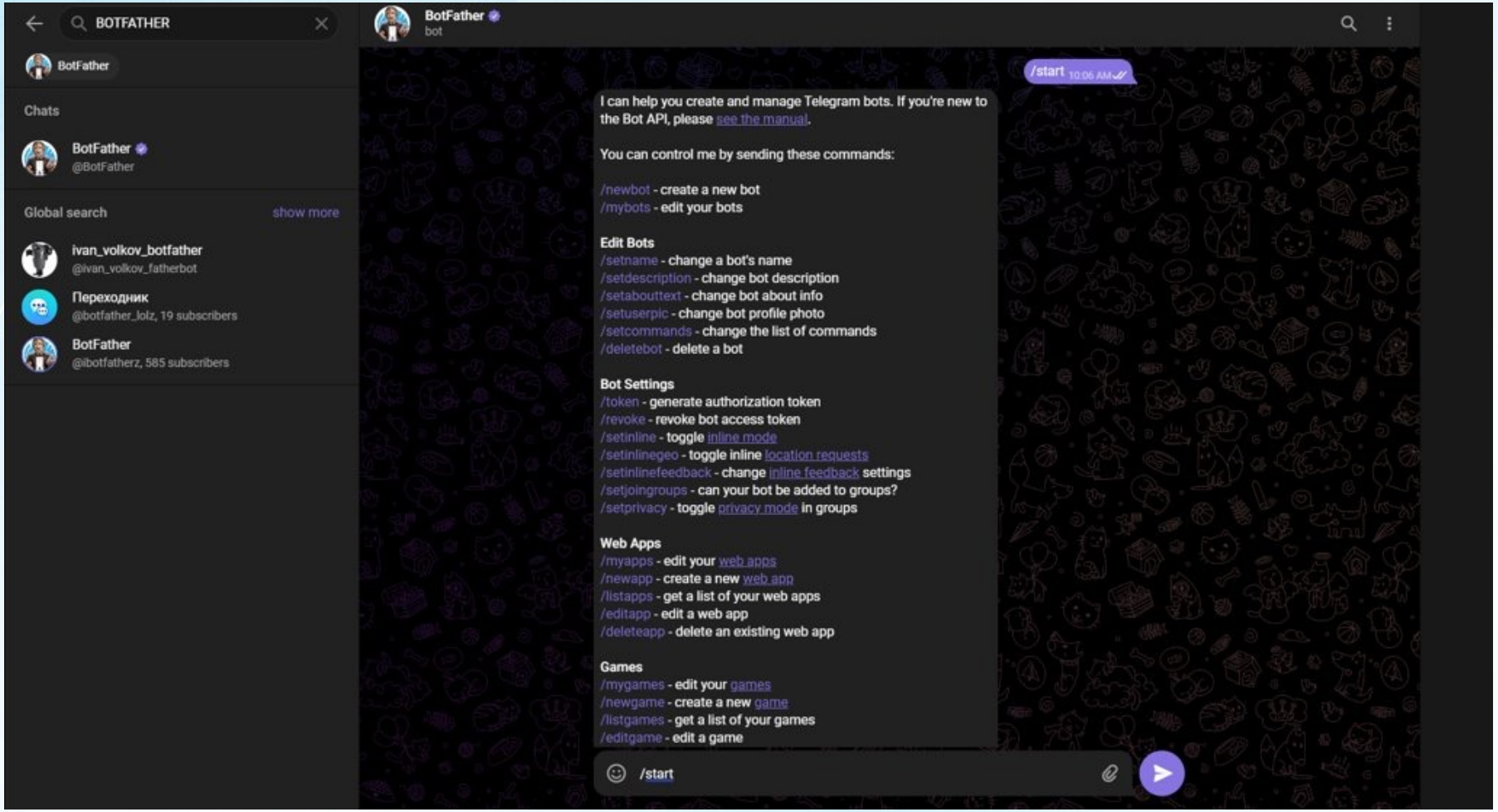
```
.idea/

**/__pycache__

.env
```

Challenge

Search for **@BotFather** inside a telegram and type **/start** to get a menu.

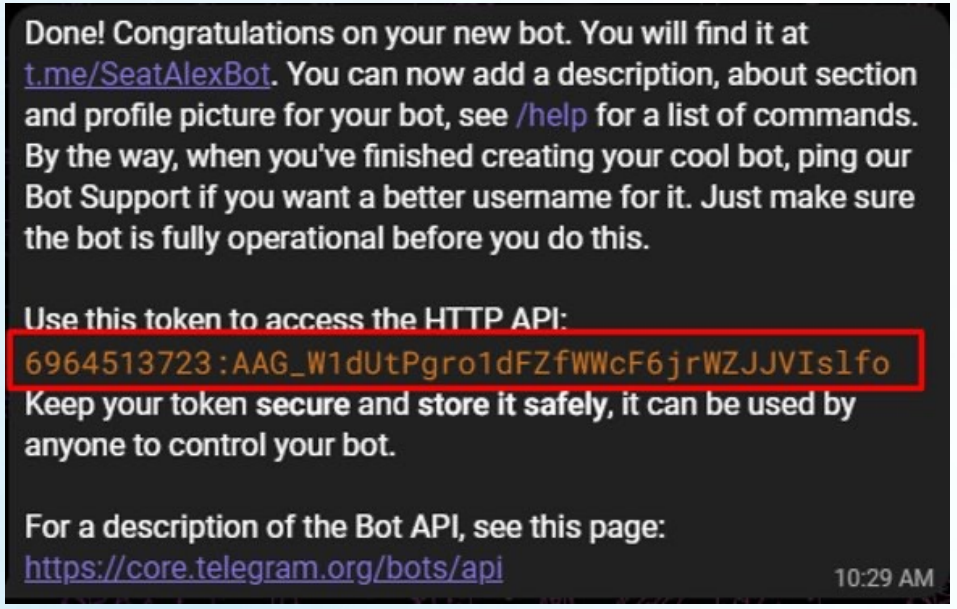
Challenge

Choose a **/newbot** option



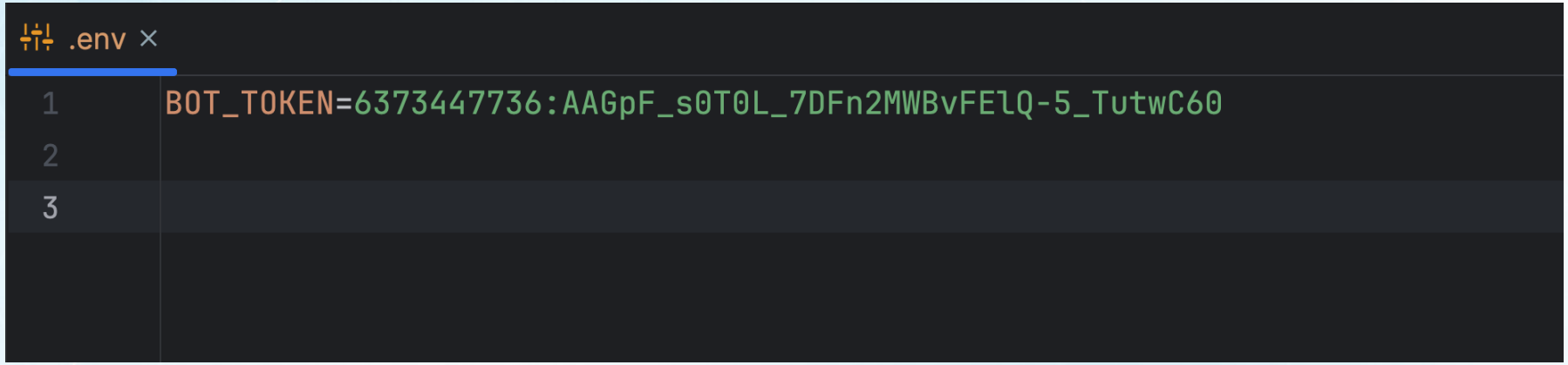
Challenge

- Choose the following name for the bot: **INT Python Challenge**
- Choose the following name for the bot: **int_python_challenge_bot**
- Make sure you're getting a message about a successful creation:



Challenge

In the project root directory create a **.env** file. Then, copy a token from the Telegram and paste a token into the **.env** file. Store it as a **BOT_TOKEN** variable like the following:



Challenge

Create a file (**main.py**) inside the project and make sure to add the following:

- Command **/start** handler
- **Any** command handler that will just send back the same message of any type. Don’t forget to include the errors handler inside a function.
- Function **main()** that will include the **INFO** logger and bot polling.
- A function **asyncio.run()** that will execute the **main()** function.
- You can take the following python file as an example.

If anything is correct, executing a file will show the following output:

```
INFO:aiogram.dispatcher:Start polling
INFO:aiogram.dispatcher:Run polling for bot @alex_komanov_bot id=6373447736 - 'INT DEMO ROBOT'
```

INT LIVE
המרכז הבינלאומי
לליסודי הייטק וחדשנות

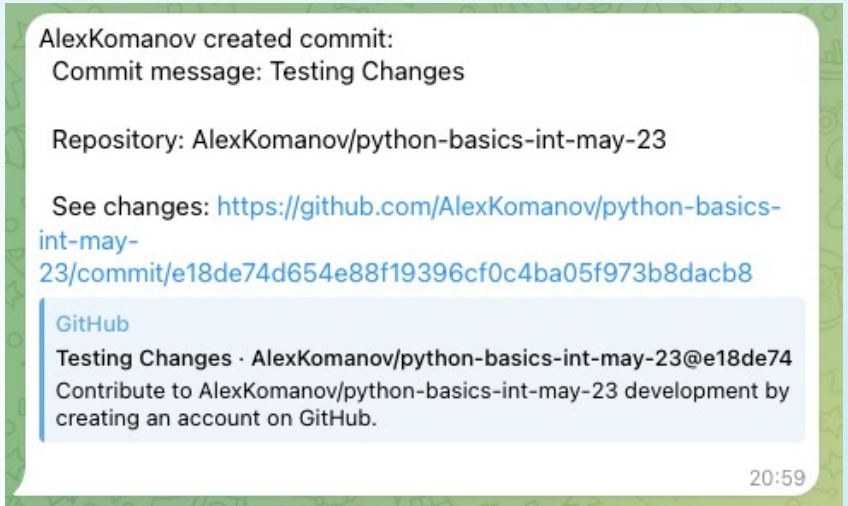
Python Challenge

Challenge

As a bonus part, you can add an integration with the **GitHub Actions Workflow**:

- See the relevant part of the lesson where it was presented.
- **Don’t forget** to create relevant **secret** variables inside a repository.
- You can use the following YAML file as an example.

If anything is correct, every push to the repository will trigger the following message:



Challenge

- You'll need to perform at least the main part. The bonus part is not a mandatory, but it is a good experience, so try to do it also.
- Add several screenshots that present how the bot is working. Create a folder inside the project and store the screenshots there.
- Commit and push the project (including the screenshots).
- Add the GitHub link (make sure that a repository is **public**) to the task inside the LMS platform and submit.