

Міністерство освіти і науки України Національний технічний університет України "КПІ імені Ігоря Сікорського "

Факультет інформатики та обчислювальної техніки Кафедра інформатики і програмної інженерії

### **3BIT**

лабораторної роботи №6 з курсу «Основи WEB - технологій»

Тема: «Створення власного API на Python (FastAPI).»

Перевірив: Виконала:

Викл. Альбрехт Й.О.

ст. Мельникова Катерина гр. IC-12

## Завдання1.

Створити web-додаток з використанням технології FastAPI, який буде працювати з інформацією відповідно вашому варіанту (Додаток 1. Таблиця 1)

## Хід роботи

Повний код проєкту можна переглянути за посиланням: https://github.com/katerynamelnykova/web-technologies-labs

Файл main.py:

```
from fastapi import FastAPI, HTTPException
app = FastAPI()
db = client["web-technologies-labs"]
collection = db["workers"]
@app.get("/")
@app.get("/workers/", response_model=List[WorkerInDB])
  workers = collection.find()
```

```
@app.get("/workers/{worker id}", response model=WorkerInDB)
def read worker(worker id: str):
      return worker serializer(worker)
  raise HTTPException(status code=404, detail="Worker not found")
@app.post("/workers/", response model=WorkerInDB)
def create worker(worker: Worker):
  worker_id = collection.insert_one(worker.model_dump()).inserted_id
@app.patch("/workers/{worker_id}", response_model=WorkerInDB)
def update worker(worker id: str, worker: WorkerUpdate):
  update data = {k: v for k, v in worker.model dump().items() if v is not None}
  if not update data:
  result = collection.update one(
      {" id": ObjectId(worker id)}, {"$set": update data}
      updated worker = collection.find one({" id": ObjectId(worker id)})
      return worker serializer(updated worker)
  raise HTTPException(status code=404, detail="Worker not found")
@app.delete("/workers/{worker id}")
def delete worker(worker id: str):
  raise HTTPException(status code=404, detail="Worker not found")
```

## workermodels.py

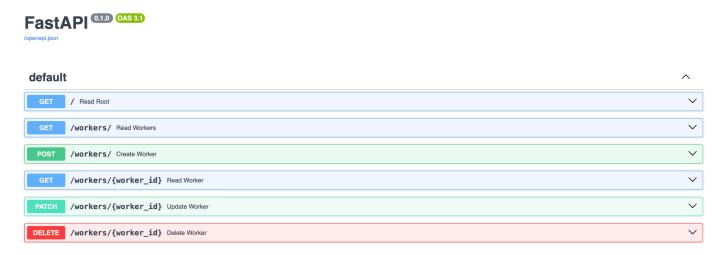
```
from pydantic import BaseModel
from typing import Optional

class Worker(BaseModel):
    name: str
    room: str
    department: str
    computer: str

class WorkerInDB(Worker):
    id: str

class WorkerUpdate(BaseModel):
    name: Optional[str] = None
    room: Optional[str] = None
    department: Optional[str] = None
    computer: Optional[str] = None
```

# Отримані результати:



```
curl -X 'GET' \
  'http://localhost:8000/workers/' \
  -H 'accept: application/json'
  http://localhost:8000/workers/
 Code
                      Details
 200
                  "1d": "obe33/0050082d3f91646d5d"
},
{
  "name": "Iващенко",
  "room": "100",
  "department": "McDook Air",
  "computer": "McDook Air",
  "id": "66e33bd650a82d9f91646a63"
                      "ta"
},
{
   "name": "Ma3yp",
   "room": "23",
   "department": "HR",
   "computer": "HP",
   "id": "66e741765cc9685707ae8231"
                                                                                                                                                                                                                                                                                                                                 Download Download
  Responses
 curl -X 'POST' \
    'http://localhost:8000/workers/' \
    -H 'accept: application/json' \
    -H 'Content-Type: application/json' \
    -d '{
        "name': "IBautyk",
        "room': "25b",
        "department": "IT",
        "computer": "Asus Vivobook Pro"
};
  Request URL
  http://localhost:8000/workers/
  Server response
  Code
  200
                       Response body
                      Download
Curl
curl -X 'GET' \
  'http://localhost:8000/workers/66e824863517a13c6596af9d' \
  -H 'accept: application/json'
Request URL
http://localhost:8000/workers/66e824863517a13c6596af9d
Server response
Code
                    Details
200
                     Response body
                     {
    "name": "Isawyk",
    "room": "Z5b",
    "department": "IT",
    "computer": "Asus Vivobook Pro",
    "id": "66e824863517a13c6596af9d"
                                                                                                                                                                                                                                                                                                                                  Download
```



#### Висновок

Під час виконання даної лабораторної роботи, використовуючи технології FastApi, я створила API для роботи з даними працівників підприємства відповідно до варіанту та виконала дослідження створених endpoints за допомогою Postman та Swagger.