

**Міністерство освіти і науки України
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
«ДНІПРОВСЬКА ПОЛІТЕХНІКА»**



Лабораторна робота №4

з дисципліни:

«Поглиблене програмування Java»

Виконав:

студент групи 123-21-2

Потоцький А. О.

Перевірив:

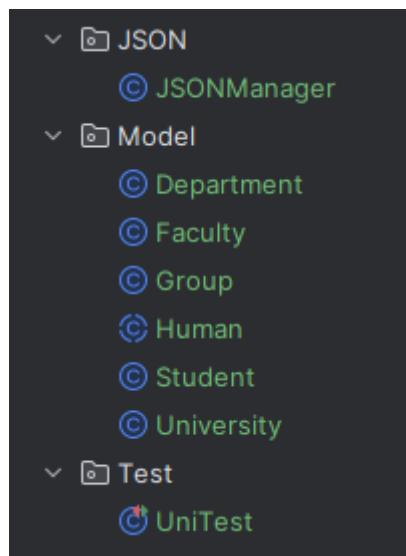
Мінесєв О.С.

м. Дніпро

2025

Лабораторна робота №4

JSON



JSONManager.java

```
package org.pototskyi.lab4.JSON;

import com.google.gson.*;
import org.pototskyi.lab4.Model.*;

import java.io.*;
import java.io.IOException;
import java.lang.reflect.Type;

public class JSONManager {
    // Ініціалізація об'єкта Gson з налаштуваннями для форматування та адаптера
    // для серіалізації/десеріалізації класу Human
    private static final Gson gson = new GsonBuilder()
        .setPrettyPrinting() // Додає красиве форматування для виводу JSON
        .registerTypeAdapter(Human.class, new HumanAdapter()) //
        // Регіструємо власний адаптер для класу Human
        .create(); // Створюємо об'єкт Gson з вказаними налаштуваннями

    // Метод для серіалізації об'єкта University в JSON формат та запису в файл
    public static void saveToFile(University university, String filename) throws
    IOException {
        // Використовуємо FileWriter для запису в файл
        try (FileWriter writer = new FileWriter(filename)) {
            // Перетворюємо об'єкт university в JSON і записуємо в файл
            gson.toJson(university, writer);
        }
    }

    // Метод для десеріалізації JSON з файлу в об'єкт University
    public static University loadFromFile(String filename) throws IOException {
        // Використовуємо FileReader для зчитування з файлу
        try (FileReader reader = new FileReader(filename)) {
            // Перетворюємо JSON з файлу в об'єкт типу University
            return gson.fromJson(reader, University.class);
        }
    }

    // Внутрішній клас, який реалізує адаптер для серіалізації та десеріалізації
    // класу Human
    private static class HumanAdapter implements JsonSerializer<Human>,
```

```

JsonDeserializer<Human> {

    // Метод для серіалізації об'єкта Human в JSON
    @Override
    public JsonElement serialize(Human src, Type typeOfSrc,
JsonSerializationContext context) {
        // Перетворюємо об'єкт в JSON об'єкт
        JsonObject jsonObject = context.serialize(src).getAsJsonObject();
        // Додаємо додаткове поле "type", яке вказує на реальний клас
        об'єкта
        jsonObject.addProperty("type", src.getClass().getSimpleName());
        return jsonObject;
    }

    // Метод для десеріалізації JSON в об'єкт Human
    @Override
    public Human deserialize(JsonElement json, Type typeOfT,
JsonDeserializationContext context) throws JsonParseException {
        // Отримуємо JSON об'єкт
        JsonObject jsonObject = json.getAsJsonObject();
        // Отримуємо значення поля "type" для визначення реального класу
        String type = jsonObject.get("type").getString();
        // Вибір класу в залежності від значення "type"
        Class<? extends Human> clazz = "Student".equals(type) ?
Student.class : Human.class;
        // Десеріалізуємо JSON в об'єкт відповідного класу
        return context.deserialize(jsonObject, clazz);
    }
}
}

```

UniTest.java

```

package org.pototskyi.lab4.Test;

import org.pototskyi.lab4.JSON.JSONManager;
import org.pototskyi.lab4.Model.*;
import org.junit.jupiter.api.Test;

import java.io.IOException;

import static org.junit.jupiter.api.Assertions.assertEquals;

public class UniTest {
    // Тестування серіалізації і десеріалізації об'єкта University в/з JSON
    // формату
    @Test
    public void testJsonSerialization() throws IOException {

        University oldUniversity = new University("Global Knowledge University",
new Student("Alex", "Peterson", "James", Human.Sex.MALE, "S90"));

        Faculty faculty1 = new Faculty("Faculty of Engineering", new
Student("Liam", "Adams", "Michael", Human.Sex.MALE, "S91"));
        Faculty faculty2 = new Faculty("Faculty of Social Sciences", new
Student("Emma", "Thompson", "Olivia", Human.Sex.FEMALE, "S92"));

        Department department11 = new Department("Artificial Intelligence and
Robotics", new Student("Sophia", "Martinez", "Elena", Human.Sex.FEMALE, "S93"));
        Department department12 = new Department("Cybersecurity and Networks",
new Student("Ethan", "Williams", "Henry", Human.Sex.MALE, "S94"));
        Department department21 = new Department("Psychology and Human
Behavior", new Student("Isabella", "Johnson", "Marie", Human.Sex.FEMALE,

```

```

"S95"));
    Department department22 = new Department("Philosophy and Ethics", new
Student("Benjamin", "Clark", "Edward", Human.Sex.MALE, "S96"));

    Group group111 = new Group("AI-201", new Student("Daniel", "Evans",
"Thomas", Human.Sex.MALE, "S97"));
    Group group112 = new Group("AI-202", new Student("Ava", "Harris",
"Grace", Human.Sex.FEMALE, "S98"));
    Group group121 = new Group("CS-301", new Student("Noah", "Baker",
"Andrew", Human.Sex.MALE, "S99"));
    Group group122 = new Group("CS-302", new Student("Mia", "Rodriguez",
"Rose", Human.Sex.FEMALE, "S100"));
    Group group211 = new Group("PSY-101", new Student("Elijah", "Gonzalez",
"James", Human.Sex.MALE, "S101"));
    Group group212 = new Group("PSY-102", new Student("Charlotte", "Nelson",
"Anne", Human.Sex.FEMALE, "S102"));
    Group group221 = new Group("PHIL-201", new Student("William", "Carter",
"Thomas", Human.Sex.MALE, "S103"));
    Group group222 = new Group("PHIL-202", new Student("Emily", "Wright",
"Lillian", Human.Sex.FEMALE, "S104"));

    group111.addStudent(new Student("Henry", "Mitchell", "David",
Human.Sex.MALE, "S105"));
    group111.addStudent(new Student("Ella", "Perez", "Sophia",
Human.Sex.FEMALE, "S106"));
    group112.addStudent(new Student("Lucas", "Roberts", "James",
Human.Sex.MALE, "S107"));
    group112.addStudent(new Student("Zoe", "Foster", "Grace",
Human.Sex.FEMALE, "S108"));
    group121.addStudent(new Student("Mason", "Diaz", "Henry",
Human.Sex.MALE, "S109"));
    group121.addStudent(new Student("Aria", "Russell", "Isabella",
Human.Sex.FEMALE, "S110"));
    group122.addStudent(new Student("Owen", "Simmons", "Benjamin",
Human.Sex.MALE, "S111"));
    group122.addStudent(new Student("Scarlett", "Patterson", "Charlotte",
Human.Sex.FEMALE, "S112"));
    group211.addStudent(new Student("Jackson", "Reed", "Elijah",
Human.Sex.MALE, "S113"));
    group211.addStudent(new Student("Grace", "Bailey", "Scarlett",
Human.Sex.FEMALE, "S114"));
    group212.addStudent(new Student("Sebastian", "Cooper", "Lucas",
Human.Sex.MALE, "S115"));
    group212.addStudent(new Student("Lily", "Morris", "Avery",
Human.Sex.FEMALE, "S116"));
    group221.addStudent(new Student("Carter", "Bell", "Daniel",
Human.Sex.MALE, "S117"));
    group221.addStudent(new Student("Natalie", "Murphy", "Evelyn",
Human.Sex.FEMALE, "S118"));
    group222.addStudent(new Student("Julian", "Howard", "Jack",
Human.Sex.MALE, "S119"));
    group222.addStudent(new Student("Victoria", "Cook", "Lillian",
Human.Sex.FEMALE, "S120"));

    department11.addGroup(group111);
    department11.addGroup(group112);
    department12.addGroup(group121);
    department12.addGroup(group122);
    department21.addGroup(group211);
    department21.addGroup(group212);
    department22.addGroup(group221);
    department22.addGroup(group222);

```

```

        faculty1.addDepartment(department11);
        faculty1.addDepartment(department12);
        faculty2.addDepartment(department21);
        faculty2.addDepartment(department22);

        oldUniversity.addFaculty(faculty1);
        oldUniversity.addFaculty(faculty2);

        JSONManager.saveToFile(oldUniversity, "university.json");

        University newUniversity = JSONManager.loadFromFile("university.json");

        System.out.println("Are universities equal? " +
oldUniversity.equals(newUniversity));

        assertEquals(oldUniversity, newUniversity);
    }
}

```

Результат

```

{
  "name": "Global Knowledge University",
  "head": {
    "studentId": "S90",
    "firstName": "Alex",
    "lastName": "Peterson",
    "patronymic": "James",
    "sex": "MALE",
    "type": "Student"
  },
  "faculties": [
    {
      "name": "Faculty of Engineering",
      "head": {
        "studentId": "S91",
        "firstName": "Liam",
        "lastName": "Adams",
        "patronymic": "Michael",
        "sex": "MALE",
        "type": "Student"
      },
      "departments": [
        {
          "name": "Artificial Intelligence and Robotics",
          "head": {
            "studentId": "S93",
            "firstName": "Sophia",
            "lastName": "Martinez",
            "patronymic": "Elena",
            "sex": "FEMALE",
            "type": "Student"
          },
          "groups": [
            {
              "name": "AI-201",
              "head": {
                "studentId": "S97",
                "firstName": "Daniel",
                "lastName": "Evans",

```

```

        "patronymic": "Thomas",
        "sex": "MALE",
        "type": "Student"
    },
    "students": [
        {
            "studentId": "S105",
            "firstName": "Henry",
            "lastName": "Mitchell",
            "patronymic": "David",
            "sex": "MALE"
        },
        {
            "studentId": "S106",
            "firstName": "Ella",
            "lastName": "Perez",
            "patronymic": "Sophia",
            "sex": "FEMALE"
        }
    ]
},
{
    "name": "AI-202",
    "head": {
        "studentId": "S98",
        "firstName": "Ava",
        "lastName": "Harris",
        "patronymic": "Grace",
        "sex": "FEMALE",
        "type": "Student"
    },
    "students": [
        {
            "studentId": "S107",
            "firstName": "Lucas",
            "lastName": "Roberts",
            "patronymic": "James",
            "sex": "MALE"
        },
        {
            "studentId": "S108",
            "firstName": "Zoe",
            "lastName": "Foster",
            "patronymic": "Grace",
            "sex": "FEMALE"
        }
    ]
}
],
{
    "name": "Cybersecurity and Networks",
    "head": {
        "studentId": "S94",
        "firstName": "Ethan",
        "lastName": "Williams",
        "patronymic": "Henry",
        "sex": "MALE",
        "type": "Student"
    },
    "groups": [
        {
            "name": "CS-301",
            "head": {
                "studentId": "S99",
                "firstName": "Noah",

```

```

        "lastName": "Baker",
        "patronymic": "Andrew",
        "sex": "MALE",
        "type": "Student"
    },
    "students": [
        {
            "studentId": "S109",
            "firstName": "Mason",
            "lastName": "Diaz",
            "patronymic": "Henry",
            "sex": "MALE"
        },
        {
            "studentId": "S110",
            "firstName": "Aria",
            "lastName": "Russell",
            "patronymic": "Isabella",
            "sex": "FEMALE"
        }
    ]
},
{
    "name": "CS-302",
    "head": {
        "studentId": "S100",
        "firstName": "Mia",
        "lastName": "Rodriguez",
        "patronymic": "Rose",
        "sex": "FEMALE",
        "type": "Student"
    },
    "students": [
        {
            "studentId": "S111",
            "firstName": "Owen",
            "lastName": "Simmons",
            "patronymic": "Benjamin",
            "sex": "MALE"
        },
        {
            "studentId": "S112",
            "firstName": "Scarlett",
            "lastName": "Patterson",
            "patronymic": "Charlotte",
            "sex": "FEMALE"
        }
    ]
}
]
}
]
},
{
    "name": "Faculty of Social Sciences",
    "head": {
        "studentId": "S92",
        "firstName": "Emma",
        "lastName": "Thompson",
        "patronymic": "Olivia",
        "sex": "FEMALE",
        "type": "Student"
    },
    "departments": [
        {
            "name": "Psychology and Human Behavior",

```

```

    "head": {
      "studentId": "S95",
      "firstName": "Isabella",
      "lastName": "Johnson",
      "patronymic": "Marie",
      "sex": "FEMALE",
      "type": "Student"
    },
    "groups": [
      {
        "name": "PSY-101",
        "head": {
          "studentId": "S101",
          "firstName": "Elijah",
          "lastName": "Gonzalez",
          "patronymic": "James",
          "sex": "MALE",
          "type": "Student"
        },
        "students": [
          {
            "studentId": "S113",
            "firstName": "Jackson",
            "lastName": "Reed",
            "patronymic": "Elijah",
            "sex": "MALE"
          },
          {
            "studentId": "S114",
            "firstName": "Grace",
            "lastName": "Bailey",
            "patronymic": "Scarlett",
            "sex": "FEMALE"
          }
        ]
      },
      {
        "name": "PSY-102",
        "head": {
          "studentId": "S102",
          "firstName": "Charlotte",
          "lastName": "Nelson",
          "patronymic": "Anne",
          "sex": "FEMALE",
          "type": "Student"
        },
        "students": [
          {
            "studentId": "S115",
            "firstName": "Sebastian",
            "lastName": "Cooper",
            "patronymic": "Lucas",
            "sex": "MALE"
          },
          {
            "studentId": "S116",
            "firstName": "Lily",
            "lastName": "Morris",
            "patronymic": "Avery",
            "sex": "FEMALE"
          }
        ]
      }
    ]
  },
  {

```



```

    "name": "Philosophy and Ethics",
    "head": {
      "studentId": "S96",
      "firstName": "Benjamin",
      "lastName": "Clark",
      "patronymic": "Edward",
      "sex": "MALE",
      "type": "Student"
    },
    "groups": [
      {
        "name": "PHIL-201",
        "head": {
          "studentId": "S103",
          "firstName": "William",
          "lastName": "Carter",
          "patronymic": "Thomas",
          "sex": "MALE",
          "type": "Student"
        },
        "students": [
          {
            "studentId": "S117",
            "firstName": "Carter",
            "lastName": "Bell",
            "patronymic": "Daniel",
            "sex": "MALE"
          },
          {
            "studentId": "S118",
            "firstName": "Natalie",
            "lastName": "Murphy",
            "patronymic": "Evelyn",
            "sex": "FEMALE"
          }
        ]
      }
    ],
    {
      "name": "PHIL-202",
      "head": {
        "studentId": "S104",
        "firstName": "Emily",
        "lastName": "Wright",
        "patronymic": "Lillian",
        "sex": "FEMALE",
        "type": "Student"
      },
      "students": [
        {
          "studentId": "S119",
          "firstName": "Julian",
          "lastName": "Howard",
          "patronymic": "Jack",
          "sex": "MALE"
        },
        {
          "studentId": "S120",
          "firstName": "Victoria",
          "lastName": "Cook",
          "patronymic": "Lillian",
          "sex": "FEMALE"
        }
      ]
    }
  ]
}

```

```
]
}
]
```