

Uniwersytet WSB Merito

Programowanie obiektowe. Zadanie 4

Imiona i nazwiska: Baranovskyi Ivan, Andrii Popchuk

Adresy email: ivan.baranovskyi@gmail.com

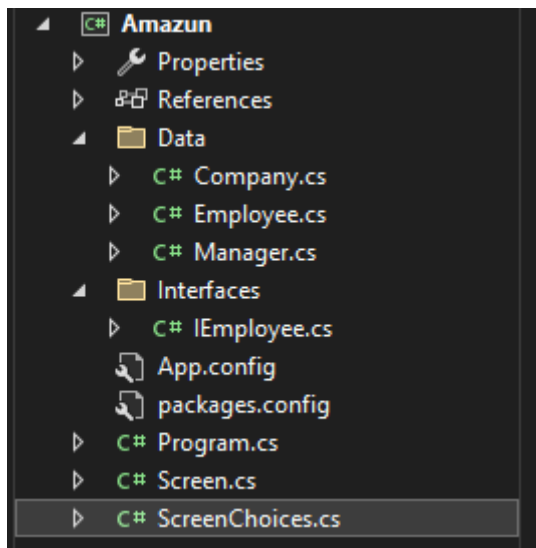
Numery albumu: **144401, 144561**

Data: 10.09.2023

Link do repo na Github: https://github.com/ivan-baranovskyi/Programowanie_obiektowe-4.git

Rozdział 1 – Zadanie 1

Wybraliśmy projekt System zarządzania pracownikami i wykonaliśmy wszystkie zadania, które zostały określone, poniżej znajdują się zrzuty ekranu z zajęć:



```
14 references
public class Employee : IEmployee
{
    8 references
    public string EmployeeId { get; set; }
    3 references
    public string Name { get; set; }
    4 references
    public string Position { get; set; }
    4 references
    public double HourlyRate { get; set; }
}
```

```

7 references
public enum ScreenChoices
{
    Exit = 0,
    AddEmployee = 1,
    RemoveEmployee = 2,
    ModifyEmployee = 3,
    CalculateSalary = 4,
    ShowEmployees = 5,
}

```

```

public class Screen
{
    private Company company = new Company();

    private string FilePath = "CompanyWorkers.json";

    1 reference
    public void Show()...

    #region Private Methods
    1 reference
    private void CalculateEmployeeSalary()...

    1 reference
    private void ShowEmployees()...

    1 reference
    private void RemoveEmployeeFromCompany()...

    1 reference
    private void AddNewEmployee()...

    #endregion // Private Methods
}

```

```

public class Manager : Employee
{
    1 reference
    public int Bonus { get; set; }
}

```

2 references

```
public class Company
{
    public List<Employee> Employees = new List<Employee>();
    4 references
    private string FilePath { get; set; } = "CompanyWorkers.json";

    /// <summary>
    /// Adds new employee
    /// </summary>
    /// <param name="newEmployee"></param>
    1 reference
    public void AddEmployee(Employee newEmployee) {...}

    /// <summary>
    /// Removes employees
    /// </summary>
    /// <param name="employeeId"></param>
    1 reference
    public void RemoveEmployee(string employeeId) {...}

    /// <summary>
    /// Calculating salary
    /// </summary>
    /// <param name="employeeId"></param>
    /// <param name="countOfHours"></param>
    1 reference
    public void CalculateSalary(string employeeId, int countOfHours) {...}

    /// <summary>
    /// Deserializing employees
    /// </summary>
    /// <param name="filePath"></param>
    /// <returns></returns>
    4 references
    public List<Employee> GetListEmployees(string filePath) {...}

    /// <summary>
    /// Serializing employees
    /// </summary>
    /// <param name="company"></param>
    2 references
    public void SerializeToJson(List<Employee> company) {...}

    /// <summary>
    /// Generating unique employee Id
    /// </summary>
    /// <returns></returns>
    1 reference
    private string GenerateUniqueEmployeeId() {...}
}
```