Sean Butler
CS 5800 Adv. Soft. Eng.
github: https://github.com/aire39/soft_eng_examples/tree/master/Assignment_1

Software Engineering Assignment 1

1. Inheritance

```
1 import java.util.Vector;
2
3 public class Main {
4     public static void main(String[] args) {
5         SalariedEmployee employee_joe = new SalariedEmployee("Joe", "Jones", "111-11-1111
  ", 2500);
6         HourlyEmployee employee_stephanie = new HourlyEmployee("Stephanie", "Smith", "222
  -22-2222", 25.0, 32);
7         HourlyEmployee employee_mary = new HourlyEmployee("Mary", "Quinn", "333-33-3333"
  , 19.0, 47);
8         CommissionEmployee employee_nicole = new CommissionEmployee("Nicole", "Dior", "
  444-44-4444 ", 15.0, 50000.0);
9         SalariedEmployee employee_renwa = new SalariedEmployee("Renwa", "Chanel", "555-55
  -5555", 1700.0);
10        BaseEmployee employee_mike = new BaseEmployee("Mike", "Davenport", "666-66-6666"
  , 95000.0);
11        CommissionEmployee employee_mahnaz = new CommissionEmployee("Mahnaz", "Vaziri", "
  777-77-7777", 22.0, 40000.0);
12
13        System.out.print(employee_joe.GetFirstName() + " " + employee_joe.GetLastName
  () + " " + employee_joe.GetSocialSecurityNumber() + " " + employee_joe.GetWeeklySalary
  () + "\n");
14        System.out.print(employee_stephanie.GetFirstName() + " " + employee_stephanie.
  GetLastName() + " " + employee_stephanie.GetSocialSecurityNumber() + " " +
  employee_stephanie.GetWage() + " " + employee_stephanie.GetNumHoursWorked() + "\n");
15        System.out.print(employee_mary.GetFirstName() + " " + employee_mary.GetLastName
  () + " " + employee_mary.GetSocialSecurityNumber() + " " + employee_mary.GetWage() + " "
   + employee_mary.GetNumHoursWorked() + "\n");
16        System.out.print(employee_nicole.GetFirstName() + " " + employee_nicole.
  GetLastName() + " " + employee_nicole.GetSocialSecurityNumber() + " " + employee_nicole.
  GetCommissionRate() + " " + employee_nicole.GetGrossSales() + "\n");
17        System.out.print(employee_renwa.GetFirstName() + " " + employee_renwa.GetLastName
  () + " " + employee_renwa.GetSocialSecurityNumber() + " " + employee_renwa.
  GetWeeklySalary() + "\n");
18        System.out.print(employee_mike.GetFirstName() + " " + employee_mike.GetLastName
  () + " " + employee_mike.GetSocialSecurityNumber() + " " + employee_mike.GetBaseSalary
  () + "\n");
19        System.out.print(employee_mahnaz.GetFirstName() + " " + employee_mahnaz.
  GetLastName() + " " + employee_mahnaz.GetSocialSecurityNumber() + " " + employee_mahnaz.
  GetCommissionRate() + " " + employee_mahnaz.GetGrossSales() + "\n");
20    }
21 }
```

```java
public abstract class Employee {
    protected String firstName = "none";
    protected String lastName = "none";
    protected String socialSecurityNumber = "xxx-xx-xxxx";

    Employee() {}
    Employee(String first_name, String last_name, String social_security_number)
    {
        SetName(first_name, last_name);
        SetSocialSecurityNumber(social_security_number);
    }

    public void SetName(String first_name, String last_name)
    {
        firstName = first_name;
        lastName = last_name;
    }

    public void SetSocialSecurityNumber(String social_security_number)
    {
        socialSecurityNumber = social_security_number;
    }

    public String GetFirstName() {
        return firstName;
    }

    public String GetLastName() {
        return lastName;
    }

    public String GetSocialSecurityNumber() {
        return socialSecurityNumber;
    }
}

```

```java
public class BaseEmployee extends Employee {
    private double baseSalary = 0.0;

    public BaseEmployee() {}
    public BaseEmployee(String first_name, String last_name, String
social_security_number, double base_salary) {
        super(first_name, last_name, social_security_number);
        SetBaseSalary(base_salary);
    }

    public void SetBaseSalary(double base_salary) {
        baseSalary = base_salary;
    }

    public double GetBaseSalary() {
        return baseSalary;
    }
}

```

```java
public class HourlyEmployee extends Employee {
    private double wage = 0.0;
    private int numberHoursWorked = 0;

    HourlyEmployee() {}
    HourlyEmployee(String first_name, String last_name, String social_security_number,
  double wage, int number_hours_worked) {
        super(first_name, last_name, social_security_number);
        SetWage(wage);
        SetNumHoursWorked(number_hours_worked);
    }

    void SetNumHoursWorked(int num_hours_worked) {
        numberHoursWorked = num_hours_worked;
    }

    int GetNumHoursWorked() {
        return numberHoursWorked;
    }

    public void SetWage(double wage) {
        this.wage = wage;
    }

    public double GetWage() {
        return wage;
    }

}

```

```java
public class SalariedEmployee extends Employee {
    private double weeklySalary = 0.0;

    SalariedEmployee() {}
    SalariedEmployee(String first_name, String last_name, String social_security_number,
  double weekly_salary) {
        super(first_name, last_name, social_security_number);
        SetWeeklySalary(weekly_salary);
    }

    public void SetWeeklySalary(double weekly_salary) {
        weeklySalary = weekly_salary;
    }

    public double GetWeeklySalary() {
        return weeklySalary;
    }
}

```

```java
public class CommissionEmployee extends Employee {
    private double commissionRate = 0.0;
    private double grossSales = 0.0;

    CommissionEmployee() {}

    CommissionEmployee(String first_name, String last_name, String social_security_number
  , double commission_rate, double gross_sales) {
        super(first_name, last_name, social_security_number);
        SetCommissionRate(commission_rate);
        SetGrossSales(gross_sales);
    }

    public void SetGrossSales(double gross_sales) {
        grossSales = gross_sales;
    }

    public double GetGrossSales() {
        return grossSales;
    }

    public void SetCommissionRate(double commission_rate) {
        commissionRate = commission_rate;
    }

    public double GetCommissionRate() {
        return commissionRate;
    }
}

```

Results

```
Joe Jones 111-11-1111 2500.0
Stephanie Smith 222-22-2222 25.0 32
Mary Quinn 333-33-3333 19.0 47
Nicole Dior 444-44-4444  15.0 50000.0
Renwa Chanel 555-55-5555 1700.0
Mike Davenport 666-66-6666 95000.0
Mahnaz Vaziri 777-77-7777 22.0 40000.0


Process finished with exit code 0
```

2. Polymorphism

```java
import java.util.Vector;

public class Main {
    public static void main(String[] args) {
        Vector<Ship> ships = new Vector<>();
        ships.add(new Ship("Basic Ship", 2023));
        ships.add(new CruiseShip("Cruise Ship", 2011, 1000));
        ships.add(new CargoShip("Cargo Ship", 2005, 50));

        for (Ship s : ships) {
            s.print();
        }
    }
}
```

```java
public class Ship {
    protected String name = "none";
    protected int year = 0;

    Ship() {}

    Ship(String name, int year) {
        this.name = name;
        this.year = year;
    }

    public void print() {
        System.out.print("Ship: " + name + " Year: " + year + "\n");
    }
}
```

```java
public class CargoShip extends Ship {
    private int cargoCapacityInTons = 0;

    CargoShip() {
        super();
    }

    CargoShip(String name, int year, int cargo_capacity_in_tons) {
        super(name, year);
        cargoCapacityInTons = cargo_capacity_in_tons;
    }

    @Override
    public void print() {
        System.out.print("Ship: " + name + " Year: " + year + " Cargo Capacity (Tons): "
    + cargoCapacityInTons + "\n");
    }
}
```

```java
public class CruiseShip extends Ship {

    private int maxPassengers = 0;

    CruiseShip() {
        super();
    }

    CruiseShip(String name, int year, int max_passengers) {
        super(name, year);
        maxPassengers = max_passengers;
    }

    @Override
    public void print() {
        System.out.print("Ship: " + name + " Year: " + year + " Max Passengers: " +
maxPassengers + "\n");
    }
}

```

Results

```
Ship: Basic Ship Year: 2023
Ship: Cruise Ship Year: 2011 Max Passengers: 1000
Ship: Cargo Ship Year: 2005 Cargo Capacity (Tons): 50

Process finished with exit code 0
```

3. Aggregation

```java
public class Book {
    public String bookTitle = "";
    public String bookAuthor = "";
    public String bookPublisher = "";
}

```

```java
public class Main {
    public static void main(String[] args) {
        CourseWork course_work = new CourseWork();
        course_work.AddCourse("Adv. Software Engineering"
                              , "Nema"
                              , "Davapanah"
                              , "3-2636"
                              , "Clean Code"
                              , "Robort C. Martin"
                              , "Pearson 1st edition");

        Course new_course = new Course();
        new_course.SetCourseName("Light Literaturature");
        new_course.AddInstructor("Donald", "Brewer", "6-1545");
        new_course.AddInstructor("Jeff", "Baloon", "6-1555");
        new_course.AddBook("Shining Light", "Brailer Nuten", "Fun House");
        new_course.AddBook("Light Text", "Jasonr Soler", "Gene Publishing");

        course_work.AddCourse(new_course);

        course_work.print();
    }
}
```

```java
import java.util.Vector;

public class Course {
    private String name = "";

    private final Vector<Instructor> instructors = new Vector<>();
    private final Vector<Book> books = new Vector<>();

    public void SetCourseName(String name) {
        this.name = name;
    }

    public String GetCourseName() {
        return name;
    }

    public void AddInstructor(String first_name, String last_name, String office_number
) {
        Instructor new_instructor = new Instructor();
        new_instructor.instructorFirstName = first_name;
        new_instructor.instructorLastName = last_name;
        new_instructor.instructorOfficeNumber = office_number;

        instructors.add(new_instructor);
    }

    public String GetInstructorFirstName(int index) {
        return instructors.get(index).instructorFirstName;
    }

    public String GetInstructorLastName(int index) {
        return instructors.get(index).instructorLastName;
    }

    public String GetInstructorOfficeNumber(int index) {
        return instructors.get(index).instructorOfficeNumber;
    }

    public void AddBook(String title, String author, String publisher) {
        Book new_book = new Book();
        new_book.bookTitle = title;
        new_book.bookAuthor = author;
        new_book.bookPublisher = publisher;

        books.add(new_book);
    }

    public String GetBookTitle(int index) {
        return books.get(index).bookTitle;
    }

    public String GetBookAuthor(int index) {
        return books.get(index).bookAuthor;
    }

    public String GetBookPublisher(int index) {
        return books.get(index).bookPublisher;
    }
    public void print()
    {
        System.out.println("Course Name: " + name);

        System.out.println("Instructor Name: ");
        for (Instructor i : instructors) {
            System.out.print(i.instructorFirstName + " " + i.instructorLastName + ", ");
        }

```

```
        System.out.println();

        for (int i=0; i<books.size(); i++) {
            System.out.println("Book " + (i+1) + ": " + books.get(i).bookTitle);
            System.out.println("Book " + (i+1) + " Author: " + books.get(i).bookAuthor);
        }
    }
}

```

```
import java.util.Vector;

public class CourseWork {
    private Vector<Course> courses = new Vector<>();

    CourseWork() {

    }

    public void AddCourse(String course_name
                          ,String instructor_first_name
                          ,String instructor_last_name
                          ,String instructor_office_number
                          ,String book_title
                          ,String book_author
                          ,String book_publisher) {
        Course add_course = new Course();

        add_course.SetCourseName(course_name);
        add_course.AddInstructor(instructor_first_name, instructor_last_name,
instructor_office_number);
        add_course.AddBook(book_title, book_author, book_publisher);

        courses.add(add_course);
    }

    public void AddCourse(Course new_course) {
        courses.add(new_course);
    }

    public void print() {
        for (Course c : courses) {
            c.print();
            System.out.println("");
            System.out.println("");
        }
    }
}

```

```
public class Instructor {
    public String instructorFirstName = "";
    public String instructorLastName = "";
    public String instructorOfficeNumber = "";
}
```

Result

```
Course Name: Adv. Software Engineering
Instructor Name:
Nema Davapanah,
Book 1: Clean Code
Book 1 Author: Robort C. Martin


Course Name: Light Literaturature
Instructor Name:
Donald Brewer, Jeff Baloon,
Book 1: Shining Light
Book 1 Author: Brailer Nuten
Book 2: Light Text
Book 2 Author: Jasonr Soler
```

4. Composition

```
1 public class File {
2     private String fileName = "";
3
4     File(String file_name) {
5         fileName = file_name;
6     }
7
8     public void SetFileName(String file_name) {
9         fileName = file_name;
10    }
11
12    public String GetFileName() {
13        return fileName;
14    }
15 }
16
```

```java
public class Main {
    public static void main(String[] args) {

        Folder php_demo1_folder = new Folder("php_demo1");

        Folder sources_folder = new Folder("Sources");
        php_demo1_folder.AddFolder(sources_folder);

        Folder phalcon_folder = new Folder(".phalcon");
        sources_folder.AddFolder(phalcon_folder);

        Folder app_folder = new Folder("app");
        sources_folder.AddFolder(app_folder);

        Folder config_folder = new Folder("config");
        app_folder.AddFolder(config_folder);

        Folder controllers_folder = new Folder("controllers");
        app_folder.AddFolder(controllers_folder);

        Folder library_folder = new Folder("library");
        app_folder.AddFolder(library_folder);

        Folder migrations_folder = new Folder("migrations");
        app_folder.AddFolder(migrations_folder);

        Folder models_folder = new Folder("models");
        app_folder.AddFolder(models_folder);

        Folder views_folder = new Folder("views");
        app_folder.AddFolder(views_folder);

        Folder cache_folder = new Folder("cache");
        sources_folder.AddFolder(cache_folder);

        Folder public_folder = new Folder("public");
        sources_folder.AddFolder(public_folder);

        public_folder.AddFile(new File("htaccess"));
        public_folder.AddFile(new File(".htrouter.php"));
        public_folder.AddFile(new File("index.html"));

        Folder include_path_folder = new Folder("Include Path");
        php_demo1_folder.AddFolder(include_path_folder);

        Folder remote_files_folder = new Folder("Remote Files");
        php_demo1_folder.AddFolder(remote_files_folder);

        System.out.println("Original Folder Structure\n
  ----------------------------------------");
        php_demo1_folder.PrintFileStructure();

        System.out.println();
        System.out.println();

        System.out.println("Folder Structure After Deleting app Folder\n
  ----------------------------------------");
        sources_folder.Delete(app_folder);
        php_demo1_folder.PrintFileStructure();

        System.out.println();
        System.out.println();

        System.out.println("Folder Structure After Deleting public Folder\n
  ----------------------------------------");
        sources_folder.Delete(public_folder);
        php_demo1_folder.PrintFileStructure();
```

```
65      }
66  }
```

```java
import java.util.Vector;

public class Folder {
    private String folderName = "";
    private Folder parentFolder = null;
    private final Vector<Folder> subFolders = new Vector<>();
    private final Vector<File> files = new Vector<>();

    static boolean initPrint = false;

    Folder() {
    }

    Folder(String folder_name) {
        folderName = folder_name;
    }

    public void SetFolderName(String folder_name) {
        folderName = folder_name;
    }

    public void AddFolder(Folder new_folder) {
        new_folder.parentFolder = this;
        subFolders.add(new_folder);
    }

    public void AddFile(File new_file) {
        files.add(new_file);
    }

    public void Delete(Folder folder) {
        subFolders.remove(folder);
    }

    public void Delete(File file) {
        files.remove(file);
    }

    public void PrintFileStructure() {
        System.out.println(folderName);
        PrintFileStructureStart();
    }
    private void PrintFileStructureStart() {
        for(Folder f : subFolders) {
            StringBuilder indentation = new StringBuilder();
            indentation.append("  ");
            Folder tmp_folder = parentFolder;

            while (tmp_folder != null) {
                indentation.append("  ");
                tmp_folder = tmp_folder.parentFolder;
            }

            System.out.println(indentation + "+" + f.folderName);

            f.PrintFileStructureStart();
        }

        for(File f : files) {
            StringBuilder indentation = new StringBuilder();
            indentation.append("  ");
            Folder tmp_folder = parentFolder;

            while (tmp_folder != null) {
```

```
68
69              System.out.println(indentation + "+" + f.GetFileName());
70          }
71      }
72
73
74
75  }
76
```

Results

```
Original Folder Structure
-----------------------------------------
php_demo1
  +Sources
    +.phalcon
    +app
      +config
      +controllers
      +library
      +migrations
      +models
      +views
    +cache
    +public
      +htaccess
      +.htrouter.php
      +index.html
  +Include Path
  +Remote Files


Folder Structure After Deleting app Folder
-----------------------------------------
php_demo1
  +Sources
    +.phalcon
    +cache
    +public
      +htaccess
      +.htrouter.php
      +index.html
  +Include Path
  +Remote Files
```

```
Folder Structure After Deleting public Folder
-----------------------------------------
php_demo1
  +Sources
    +.phalcon
    +cache
  +Include Path
  +Remote Files

Process finished with exit code 0
```