МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ

Лабораторная работа 5
По дисциплине “СПП”

Выполнил: Вальчук А.А.
Проверил: Крощенко А.А.

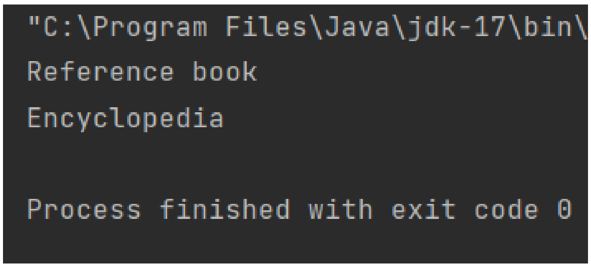
Брест 2021

Цель работы: приобрести практические навыки в области объектно-ориентированного проектирования.

Задание 1. Абстрактный класс Книга (Шифр, Автор, Название, Год, Издательство). Подклассы Справочник и Энциклопедия.

```
public abstract class Book {
   protected String cipher;
   protected String author;
   protected String title;
   protected Short year;
   protected String publisher;

   public Book() {}

   public Book(String cipher, String author, String title, Short year, String publisher) {
      this.cipher = cipher;
      this.author = author;
      this.title = title;
      this.year = year;
      this.publisher = publisher;
   }

   public String getCipher() {
      return cipher;
   }

   public String getAuthor() {
      return author;
   }

   public String getTitle() {
      return title;
   }

   public Short getYear() {
      return year;
   }

   public String getPublisher() {
      return publisher;
   }

   public void setCipher(String cipher) {
      this.cipher = cipher;
   }

   public void setAuthor(String author) {
      this.author = author;
   }

   public void setTitle(String title) {
        this.title = title;
    }

    public void setYear(Short year) {
        this.year = year;
    }

    public void setPublisher(String publisher) {
        this.publisher = publisher;
    }
}

public class Encyclopedia extends Book {
    private Integer numberOfPages;

    public Encyclopedia() {}

     public Encyclopedia(String cipher, String author, String title, Short year, String publisher, Integer
numberOfPages) {
        super(cipher, author, title, year, publisher);
        this.numberOfPages = numberOfPages;
    }

    public Integer getNumberOfPages() {
        return numberOfPages;
    }

    public void setNumberOfPages(Integer numberOfPages) {
        this.numberOfPages = numberOfPages;
    }
}

public class ReferenceBook extends Book {
    private Double price;

    public ReferenceBook() {}

      public ReferenceBook(String cipher, String author, String title, Short year, String publisher,
Double price) {
        super(cipher, author, title, year, publisher);
        this.price = price;
    }

    public Double getPrice() {
        return price;
    }

    public void setPrice(Double price) {
        this.price = price;
    }
}
```

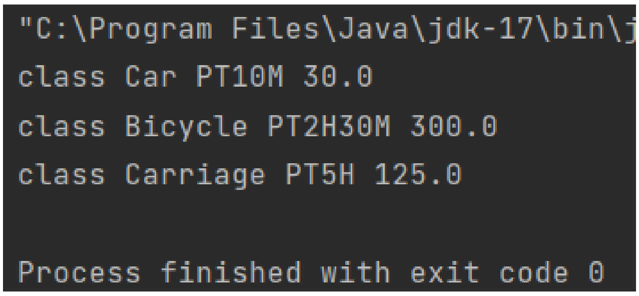
```
public class Task1 {

    public static void main(String[] args) {
                ReferenceBook referenceBook = new ReferenceBook();
        referenceBook.setTitle("Reference book");

        Encyclopedia encyclopedia = new Encyclopedia();
        encyclopedia.setTitle("Encyclopedia");

        List<Book> books = List.of(referenceBook, encyclopedia);

        for (Book book: books) {
            System.out.println(book.getTitle());
        }
    }
}
```



```
"C:\Program Files\Java\jdk-17\bin\
Reference book
Encyclopedia

Process finished with exit code 0
```

Задание 2. Создать суперкласс Транспортное средство и подклассы Автомобиль, Велосипед, Повозка. Подсчитать время и стоимость перевозки пассажиров и грузов каждым транспортным средством.

```
public class Bicycle implements IVehicle {
    @Override
    public Duration getTransportationTime(int distanceKm) {
        return Duration.ofMinutes(15).multipliedBy(distanceKm);
    }

    @Override
    public Double getTransportationCost(int distanceKm) {
        return 30.0 * distanceKm;
    }
}

public class Car implements IVehicle {
    @Override
    public Duration getTransportationTime(int distanceKm) {
        return Duration.ofMinutes(1).multipliedBy(distanceKm);
    }

    @Override
    public Double getTransportationCost(int distanceKm) {
        return 3.0 * distanceKm;
    }
}

public class Carriage implements IVehicle {
   @Override
   public Duration getTransportationTime(int distanceKm) {
      return Duration.ofMinutes(30).multipliedBy(distanceKm);
   }

   @Override
   public Double getTransportationCost(int distanceKm) {
      return 12.5 * distanceKm;
   }
}

public interface IVehicle {
   Duration getTransportationTime(int distanceKm);
   Double getTransportationCost(int distanceKm);
}

public class Task2 {

   public static void main(String[] args) {
      List<IVehicle> vehicles = List.of(new Car(), new Bicycle(), new Carriage());

      for (IVehicle vehicle: vehicles) {
         System.out.println(vehicle.getClass() + " " + vehicle.getTransportationTime(10) +
               " " + vehicle.getTransportationCost(10));
      }
   }
}
```



```
"C:\Program Files\Java\jdk-17\bin\j
class Car PT10M 30.0
class Bicycle PT2H30M 300.0
class Carriage PT5H 125.0

Process finished with exit code 0
```