package com.company;  
  
import java.util.Scanner;  
  
public class Main {  
 private static Queries *query* = new Queries();  
 private static Employee *employee* = new Employee();  
  
 public static void main(String[] args) {  
 System.*out*.println("Welcome to project management system!"); // program stars from here  
 *table\_menu*(); // calls table\_menu method  
 }  
  
 public static void table\_menu() {  
 Scanner scan = new Scanner(System.*in*);  
 System.*out*.println("1. Choose a project");  
 System.*out*.println("2. Add a project");  
 System.*out*.println("3. Exit");  
 System.*out*.println("-> Enter choice index:");  
 int choice = scan.nextInt();  
 switch (choice){ // switch used to manage the menu depending on user's choice  
 case 1: *choose\_table*(); // special method calls for each choice  
 case 2: *add\_table*();  
 case 3: System.*exit*(1);  
 }  
 }  
  
 public static void choose\_table() {  
 Scanner scan = new Scanner(System.*in*);  
  
 System.*out*.println(" Available projects:");  
 *query*.show\_tables(); // instance query calls method show\_tables which connects to database and outputs available tables  
 System.*out*.println("-> Choose a project by its name:");  
 String table\_name = scan.nextLine();  
 *query*.setChosen\_table(table\_name); // sets table name for query instance  
 if(*query*.check\_connection()) { // checks the connection with table's name  
 *manage\_menu*();  
 }  
 else{  
 System.*out*.println("This project do not exist!");  
  
 System.*out*.println("Press enter to continue...");  
 scan.nextLine();  
 scan.nextLine();  
  
 *choose\_table*();  
 }  
 }  
  
 public static void add\_table(){  
 Scanner scan = new Scanner(System.*in*);  
 System.*out*.println("-> Enter project name:");  
 String name = scan.nextLine();  
 *query*.setChosen\_table(name);  
 *query*.add\_project(name);  
  
 System.*out*.println("Press enter to continue...");  
 scan.nextLine();  
 scan.nextLine();  
  
 *manage\_menu*();  
 }  
  
 public static void remove\_table(){  
 Scanner scan = new Scanner(System.*in*);  
 *query*.remove\_project(*query*.getChosen\_table());  
  
 System.*out*.println("Press enter to continue...");  
 scan.nextLine();  
 scan.nextLine();  
  
 *table\_menu*();  
 }  
  
 public static void manage\_menu(){  
 Scanner scan = new Scanner(System.*in*);  
 System.*out*.println("Project name -> "+ *query*.getChosen\_table()+" <-");  
 System.*out*.println("1. Add employee");  
 System.*out*.println("2. Output employees");  
 System.*out*.println("3. Update employee by id");  
 System.*out*.println("4. Delete employee by id");  
 System.*out*.println("5. Calculate project total cost");  
 System.*out*.println("6. Delete this project");  
 System.*out*.println("7. Go back");  
 System.*out*.println("Enter choice index:");  
 int choice = scan.nextInt();  
 switch (choice){  
 case 1: *add\_row*();  
 case 2: *output*();  
 case 3: *update*();  
 case 4: *delete*();  
 case 5: *calculate*();  
 case 6: *remove\_table*();  
 case 7: *table\_menu*();  
 }  
 }  
  
 public static void add\_row(){  
 Scanner scan = new Scanner(System.*in*);  
  
 System.*out*.println("-> Choose position:");  
 System.*out*.println("1. Project manager");  
 System.*out*.println("2. Administrator");  
 System.*out*.println("3. Software engineer");  
 System.*out*.println("4. Assistant");  
 int choice = scan.nextInt();  
  
 System.*out*.println("-> Name of employee:");  
 scan.nextLine();  
 String name = scan.nextLine();  
  
 System.*out*.println("-> Surname of employee:");  
 String surname = scan.nextLine();  
  
 System.*out*.println("-> Salary (per month)");  
 int salary = scan.nextInt();  
  
 switch (choice){  
 case 1: *employee* = new Project\_Manager(salary, name, surname);  
 break;  
 case 2: *employee* = new Administrator(salary, name, surname);  
 break;  
 case 3: *employee* = new Software\_Engineer(salary, name, surname);  
 break;  
 case 4: *employee* = new Assistant(salary, name, surname);  
 break;  
 }  
  
 *query*.add\_row(*query*.getChosen\_table(), *employee*.getName(), *employee*.getSurname(), *employee*.getPosition(), *employee*.getSalary());  
  
 System.*out*.println("Press enter to continue...");  
 scan.nextLine();  
 scan.nextLine();  
  
 *manage\_menu*();  
 }  
  
 public static void output(){  
 Scanner scan = new Scanner(System.*in*);  
  
 System.*out*.println("-> Choose an option:");  
 System.*out*.println("1. Output every row");  
 System.*out*.println("2. Output by ID");  
  
 int choice = scan.nextInt();  
 if (choice == 1){  
 *query*.show\_table(*query*.getChosen\_table());  
 }  
 else{  
 System.*out*.println("Input row ID");  
 choice = scan.nextInt();  
 *query*.show\_row(*query*.getChosen\_table(), choice);  
 }  
  
 System.*out*.println("Press enter to continue...");  
 scan.nextLine();  
 scan.nextLine();  
  
 *manage\_menu*();  
 }  
  
 public static void update(){  
 Scanner scan = new Scanner(System.*in*);  
  
 System.*out*.println("-> Input an ID:");  
 int id = scan.nextInt();  
  
 System.*out*.println("-> Choose position:");  
 System.*out*.println("1. Project manager");  
 System.*out*.println("2. Administrator");  
 System.*out*.println("3. Software engineer");  
 System.*out*.println("4. Assistant");  
 int choice = scan.nextInt();  
  
 System.*out*.println("-> Name of employee:");  
 scan.nextLine();  
 String name = scan.nextLine();  
  
 System.*out*.println("-> Surname of employee:");  
 String surname = scan.nextLine();  
  
 System.*out*.println("-> Salary ($ per month)");  
 int salary = scan.nextInt();  
  
 switch (choice){  
 case 1: *employee* = new Project\_Manager(salary, name, surname);  
 break;  
 case 2: *employee* = new Administrator(salary, name, surname);  
 break;  
 case 3: *employee* = new Software\_Engineer(salary, name, surname);  
 break;  
 case 4: *employee* = new Assistant(salary, name, surname);  
 break;  
 }  
  
 *query*.edit\_row(*query*.getChosen\_table(), id, *employee*.getName(), *employee*.getSurname(), *employee*.getPosition(), *employee*.getSalary());  
  
 System.*out*.println("Press enter to continue...");  
 scan.nextLine();  
 scan.nextLine();  
  
 *manage\_menu*();  
 }  
  
 public static void delete(){  
 Scanner scan = new Scanner(System.*in*);  
  
 System.*out*.println("-> Input an ID:");  
 int id = scan.nextInt();  
  
 *query*.remove\_row(*query*.getChosen\_table(), id);  
 System.*out*.println("Press enter to continue...");  
 scan.nextLine();  
 scan.nextLine();  
  
 *manage\_menu*();  
 }  
  
 public static void calculate(){  
 Scanner scan = new Scanner(System.*in*);  
  
 System.*out*.println("-> Input period of time (in months)");  
 float month = (float) scan.nextDouble();  
 System.*out*.println("Total cost of the project in "+month+" months = " + month \* (float)*query*.budget(*query*.getChosen\_table()) + "$");  
 System.*out*.println("Press enter to continue...");  
 scan.nextLine();  
 scan.nextLine();  
 *manage\_menu*();  
 }  
 }

package com.company;  
  
import java.sql.\*;  
  
  
public class Queries {  
 private String chosen\_table;  
  
 public void setChosen\_table(String name) {  
 chosen\_table = name;  
 }  
  
 public String getChosen\_table() {  
 return chosen\_table;  
 }  
  
 public boolean check\_connection() {  
 String connectionUrl = "jdbc:postgresql://localhost/Projects";  
 Connection con = null;  
 Statement stmt = null;  
 try {  
 Class.*forName*("org.postgresql.Driver");  
 con = DriverManager.*getConnection*(connectionUrl, "postgres", "230801");  
 stmt = con.createStatement();  
 stmt.executeQuery("select 1 from " + chosen\_table);  
 System.*out*.println("Available tables:");  
 } catch (Exception e) {  
 return false;  
 } finally {  
 try {  
 stmt.close();  
 con.close();  
 } catch (SQLException throwables) {  
 throwables.printStackTrace();  
 }  
 }  
 return true;  
 }  
  
 public void show\_tables() {  
 String connectionUrl = "jdbc:postgresql://localhost/Projects";  
 Connection con = null;  
 ResultSet rs = null;  
 Statement stmt = null;  
 try {  
 Class.*forName*("org.postgresql.Driver");  
 con = DriverManager.*getConnection*(connectionUrl, "postgres", "230801");  
 stmt = con.createStatement();  
 rs = stmt.executeQuery("SELECT \*\n" +  
 "FROM pg\_catalog.pg\_tables\n" +  
 "WHERE schemaname != 'pg\_catalog' AND \n" +  
 " schemaname != 'information\_schema';");  
 while (rs.next())  
 System.*out*.println("\* " + rs.getString("tablename") + " \*");  
 } catch (Exception e) {  
 System.*out*.println(e);  
 } finally {  
 try {  
 rs.close();  
 stmt.close();  
 con.close();  
 } catch (SQLException throwables) {  
 throwables.printStackTrace();  
 }  
 }  
 }  
  
 public void add\_project(String name) {  
 String connectionUrl = "jdbc:postgresql://localhost/Projects";  
 Connection con = null;  
 Statement stmt = null;  
 try {  
 Class.*forName*("org.postgresql.Driver");  
 con = DriverManager.*getConnection*(connectionUrl, "postgres", "230801");  
 stmt = con.createStatement();  
 stmt.executeQuery("create table " + name + "(id serial,\n" +  
 " firstname varchar not null,\n" +  
 " lastname varchar not null,\n" +  
 " position varchar not null,\n" +  
 " salary int not null)");  
 } catch (Exception e) {  
 System.*out*.println("The project was successfully created!");  
 } finally {  
 try {  
 stmt.close();  
 con.close();  
 } catch (SQLException throwables) {  
 throwables.printStackTrace();  
 }  
 }  
 }  
  
 public void remove\_project(String table) {  
 String connectionUrl = "jdbc:postgresql://localhost/Projects";  
 Connection con = null;  
 Statement stmt = null;  
 try {  
 Class.*forName*("org.postgresql.Driver");  
 con = DriverManager.*getConnection*(connectionUrl, "postgres", "230801");  
 stmt = con.createStatement();  
 stmt.executeQuery("drop table " + table);  
 } catch (Exception e) {  
 System.*out*.println("The project was successfully deleted!");  
 } finally {  
 try {  
 stmt.close();  
 con.close();  
 } catch (SQLException throwables) {  
 throwables.printStackTrace();  
 }  
 }  
 }  
  
 public void add\_row(String table, String firstname, String lastname, String position, int salary) {  
 String connectionUrl = "jdbc:postgresql://localhost/Projects";  
 Connection con = null;  
 Statement stmt = null;  
 try {  
 Class.*forName*("org.postgresql.Driver");  
 con = DriverManager.*getConnection*(connectionUrl, "postgres", "230801");  
 stmt = con.createStatement();  
 stmt.executeQuery("insert into " + table + " values (default, '" + firstname + "', '" + lastname + "', '" + position + "', '" + salary + "')");  
 } catch (Exception e) {  
 System.*out*.println("The row was successfully created!");  
 } finally {  
 try {  
 stmt.close();  
 con.close();  
 } catch (SQLException throwables) {  
 throwables.printStackTrace();  
 }  
 }  
 }  
  
 public void remove\_row(String table, int id) {  
 String connectionUrl = "jdbc:postgresql://localhost/Projects";  
 Connection con = null;  
 Statement stmt = null;  
 try {  
 Class.*forName*("org.postgresql.Driver");  
 con = DriverManager.*getConnection*(connectionUrl, "postgres", "230801");  
 stmt = con.createStatement();  
 stmt.executeQuery("delete from " + table + " where id = " + id);  
 } catch (Exception e) {  
 System.*out*.println("The row was successfully deleted!");  
 } finally {  
 try {  
 stmt.close();  
 con.close();  
 } catch (SQLException throwables) {  
 throwables.printStackTrace();  
 }  
 }  
 }  
  
 public void edit\_row(String table, int id, String firstname, String lastname, String position, int salary) {  
 String connectionUrl = "jdbc:postgresql://localhost/Projects";  
 Connection con = null;  
 Statement stmt = null;  
 try {  
 Class.*forName*("org.postgresql.Driver");  
 con = DriverManager.*getConnection*(connectionUrl, "postgres", "230801");  
 stmt = con.createStatement();  
 stmt.executeQuery("update " + table + " set firstname = '" + firstname + "', lastname = '" + lastname + "', position = '" + position + "', salary = '" + salary + "' where id = " + id);  
 } catch (Exception e) {  
 System.*out*.println("Values with id = " + id + " were successfully edited!");  
 } finally {  
 try {  
 stmt.close();  
 con.close();  
 } catch (SQLException throwables) {  
 throwables.printStackTrace();  
 }  
 }  
  
 }  
  
 public void show\_table(String table) {  
 String connectionUrl = "jdbc:postgresql://localhost/Projects";  
 Connection con = null;  
 ResultSet rs = null;  
 Statement stmt = null;  
 try {  
 Class.*forName*("org.postgresql.Driver");  
 con = DriverManager.*getConnection*(connectionUrl, "postgres", "230801");  
 stmt = con.createStatement();  
 rs = stmt.executeQuery("select \* from " + table);  
 System.*out*.println("ID Name Surname Position Salary");  
 while (rs.next())  
 System.*out*.println("|" + rs.getInt("id") + "| |" + rs.getString("firstname") + "| |" + rs.getString("lastname") + "| |" + rs.getString("position") + "| |" + rs.getString("salary") + "|");  
 } catch (Exception e) {  
 System.*out*.println(e);  
 } finally {  
 try {  
 rs.close();  
 stmt.close();  
 con.close();  
 } catch (SQLException throwables) {  
 throwables.printStackTrace();  
 }  
 }  
 }  
  
 public void show\_row(String table, int id) {  
 String connectionUrl = "jdbc:postgresql://localhost/Projects";  
 Connection con = null;  
 ResultSet rs = null;  
 Statement stmt = null;  
 try {  
 Class.*forName*("org.postgresql.Driver");  
 con = DriverManager.*getConnection*(connectionUrl, "postgres", "230801");  
 stmt = con.createStatement();  
 rs = stmt.executeQuery("select \* from " + table + " where id = " + id);  
 System.*out*.println("ID Name Surname Position Salary");  
 while (rs.next())  
 System.*out*.println("|" + rs.getInt("id") + "| |" + rs.getString("firstname") + "| |" + rs.getString("lastname") + "| |" + rs.getString("position") + "| |" + rs.getString("salary") + "|");  
 } catch (Exception e) {  
 System.*out*.println(e);  
 } finally {  
 try {  
 rs.close();  
 stmt.close();  
 con.close();  
 } catch (SQLException throwables) {  
 throwables.printStackTrace();  
 }  
 }  
 }  
  
 public int budget(String table) {  
 String connectionUrl = "jdbc:postgresql://localhost/Projects";  
 Connection con = null;  
 ResultSet rs = null;  
 Statement stmt = null;  
 int sum = 0;  
 try {  
 Class.*forName*("org.postgresql.Driver");  
 con = DriverManager.*getConnection*(connectionUrl, "postgres", "230801");  
 stmt = con.createStatement();  
 rs = stmt.executeQuery("select salary from " + table);  
 while (rs.next())  
 sum += rs.getInt("salary");  
 } catch (Exception e) {  
 System.*out*.println(e);  
 } finally {  
 try {  
 rs.close();  
 stmt.close();  
 con.close();  
 } catch (SQLException throwables) {  
 throwables.printStackTrace();  
 }  
 return sum;  
 }  
 }  
}

package com.company;  
  
public class Project\_Manager extends Administrator implements Interface{  
  
  
 Project\_Manager(){  
  
 }  
  
 Project\_Manager(int salary, String name, String surname){  
 this.salary = salary;  
 this.name = name;  
 this.surname = surname;  
 position = "Project manager";  
 }  
  
 Project\_Manager(String name, String surname){  
 this.salary = 25000;  
 this.name = name;  
 this.surname = surname;  
 position = "Project manager";  
 }  
  
  
 @Override  
 public String SomeAction(){  
 return super.SomeAction() + " and planning";  
 }  
}

package com.company;  
  
public class Administrator extends Software\_Engineer implements Interface{  
  
  
 Administrator(){  
  
 }  
  
 Administrator(int salary, String name, String surname){  
 this.salary = salary;  
 this.name = name;  
 this.surname = surname;  
 position = "Administrator";  
 }  
  
 Administrator(String name, String surname){  
 this.salary = 20000;  
 this.name = name;  
 this.surname = surname;  
 position = "Administrator";  
 }  
  
  
 @Override  
 public String SomeAction() {  
 return "Some managing";  
 }  
}

package com.company;  
  
public class Software\_Engineer extends Assistant implements Interface{  
  
 Software\_Engineer(){  
  
 }  
  
 Software\_Engineer(int salary, String name, String surname){  
 this.salary = salary;  
 this.name = name;  
 this.surname = surname;  
 position = "Software engineer";  
  
 }  
  
 Software\_Engineer(String name, String surname){  
 salary = 15000;  
 this.name = name;  
 this.surname = surname;  
 position = "Software engineer";  
  
 }  
  
  
 @Override  
 public String SomeAction(){  
 return super.SomeAction() + " and programming";  
 }  
}

package com.company;  
  
public class Assistant extends Employee implements Interface {  
  
 Assistant(){  
  
 }  
  
 Assistant(int salary, String name, String surname){  
 this.salary = salary;  
 this.name = name;  
 this.surname = surname;  
 position = "Assistant";  
 }  
  
 Assistant(String name, String surname){  
 this.salary = 8000;  
 this.name = name;  
 this.surname = surname;  
 position = "Assistant";  
 }  
  
  
 @Override  
 public String SomeAction() {  
 return "Some assist";  
 }  
}

package com.company;  
  
public class Employee {  
 protected int salary;  
 protected String name;  
 protected String surname;  
 protected String position;  
  
 public void setSalary(int salary){  
 this.salary = salary;  
 }  
  
 public int getSalary(){  
 return salary;  
 }  
  
 public void setName(String name){  
 this.name = name;  
 }  
  
 public String getName(){  
 return name;  
 }  
  
 public void setSurname(String surname){  
 this.surname = surname;  
 }  
  
 public String getSurname(){  
 return surname;  
 }  
  
 public String getPosition(){  
 return position;  
 }  
}

package com.company;  
  
public interface Interface {  
 abstract String SomeAction();  
}