|  |  |  |  |
| --- | --- | --- | --- |
| 과목명 | 데이터베이스 | 시행일자 | 2022.04.19 (수행시간 2~3시간) |
| 점수 |  | 제출자 | 강현규 |
| 아래 요구 사항에 맞는 제품을 제작하여 제출하시오   |  | | --- | | [요구사항]  성적관리 프로그램을 제작 의뢰합니다(ver 0.9.0)  각과목은국어, 영어, 수학로 3과목입니다  학생의 정보는 학번과 이름, 각 과목의 성적, 평균점수(소수점 두자리)를 출력해주세요  입력과 수정, 삭제가 가능하도록 기능해야 합니다  콘솔 실행이며 입력시 잘못된 입력으로 인한 문제 발생을 최소화해야 합니다  프로그램은 종료후 재시작 후에도 계속 데이터 유지되어야 하며 정보는 데이터베이스 서버에서 관리되어야 합니다  사용자의 편의성을 최대한 고려해 줄 것을 부탁드립니다  실행은 마우스 더블클릭을 통해 프로그램이 실행될 수 있도록 합니다 |  |  |  |  |  | | --- | --- | --- | --- | | 배점표 | | | | | 1 | 일정시한 엄수 | 20 |  | | 2 | 사용자 편의성(UI) | 5 |  | | 3 | 문서화 | 5 |  | | 4 | 코드 가독성 | 5 |  | | 5 | 리펙토링 | 5 |  | | 6 | 문법 활용 능력 | 5 |  | | 7 | java 제약조건 | 10 |  | | 8 | 데이터베이스 연동 | 20 |  | | 9 | 무결성 제약조건 | 20 |  | | 10 | 기타 | 5 |  | | 계 | | 100 |  | | | | |
| 제품어필 | | | |
| 1. 보고싶은 학번을 입력하면 입력한 학번 성적을 볼수 있습니다.  2. 전체 성적을 삭제할 수 있습니다. | | | |
| 출력결과 | | | |
| <스크린샷>  1. 더블클릭 실행    2. 입력기능 , 전체 성적 출력    3. 학번 입력후 해당 학번 성적출력    4. 학번 입력 후 해당 학번 성적 수정    5. 학번 입력 후 해당 학번 성적 삭제    6. 학생 성적 전체 삭제 | | | |
| 소소코드 | | | |
| <첨부 1 - 실행 스크립트>    java -jar ./studentprogram.jar  <첨부 2 - DB Schema>  CREATE TABLE student (  num int NOT NULL,  name varchar(45) NOT NULL,  kor int NOT NULL,  eng int NOT NULL,  math int NOT NULL,  PRIMARY KEY (`num`)  ) ;      <첨부 3 - java code>  package com.bit.test;  import java.util.\*;  import java.sql.\*;  class Student {  int num;  String name;  int kor, eng, math;  }  public class Main {  String driver = "com.mysql.cj.jdbc.Driver";  String url = "jdbc:mysql://localhost:3306/test01";  String user = "user03";  String password = "1234";  Connection conn = null;  Statement stmt = null;  ResultSet rs = null;  public void print(String sql) {  try {  Class.forName(driver);  conn = DriverManager.getConnection(url, user, password);  stmt = conn.createStatement();  rs = stmt.executeQuery(sql);  System.out.println("---------------------------------------------");  System.out.println("학번\t이름\t국어\t영어\t수학\t평균");  System.out.println("---------------------------------------------");  while (rs.next()) {  int num = rs.getInt(1);  String name = rs.getString(2);  int kor = rs.getInt(3);  int eng = rs.getInt(4);  int math = rs.getInt(5);  double avg = (kor + eng + math) \* 100 / 3 / 100.0;  System.out.println(num + "\t" + name + "\t" + kor + "\t" + eng + "\t" + math + "\t" + avg);  }  System.out.println("---------------------------------------------");  } catch (SQLException e) {  System.out.println("sql 에러 입니다.");  e.printStackTrace();  } catch (ClassNotFoundException e) {  System.out.println("class 에러 입니다");  e.printStackTrace();  } finally {  try {  if (rs != null)  rs.close();  if (stmt != null)  stmt.close();  if (conn != null)  conn.close();  } catch (SQLException e) {  System.out.println("sql 에러 입니다.");  e.printStackTrace();  }  }  }  public void write(String sql) {  try {  Class.forName(driver);  conn = DriverManager.getConnection(url, user, password);  stmt = conn.createStatement();  stmt.executeUpdate(sql);  } catch (SQLException e) {  System.out.println("sql 에러 입니다.");  e.printStackTrace();  } catch (ClassNotFoundException e) {  System.out.println("class 에러 입니다");  e.printStackTrace();  } finally {  try {  if (stmt != null)  stmt.close();  if (conn != null)  conn.close();  } catch (SQLException e) {  System.out.println("sql 에러 입니다.");  e.printStackTrace();  }  }  }  public void modify(String sql) {  try {  Class.forName(driver);  conn = DriverManager.getConnection(url, user, password);  stmt = conn.createStatement();  stmt.executeUpdate(sql);  } catch (SQLException e) {  System.out.println("sql 에러 입니다.");  e.printStackTrace();  } catch (ClassNotFoundException e) {  System.out.println("class 에러 입니다");  e.printStackTrace();  } finally {  try {  if (stmt != null)  stmt.close();  if (conn != null)  conn.close();  } catch (SQLException e) {  System.out.println("sql 에러 입니다.");  e.printStackTrace();  }  }  }  public void delete(String sql) {  try {  Class.forName(driver);  conn = DriverManager.getConnection(url, user, password);  stmt = conn.createStatement();  stmt.executeUpdate(sql);  } catch (SQLException e) {  System.out.println("sql 에러 입니다.");  e.printStackTrace();  } catch (ClassNotFoundException e) {  System.out.println("class 에러 입니다");  e.printStackTrace();  } finally {  try {  if (stmt != null)  stmt.close();  if (conn != null)  conn.close();  } catch (SQLException e) {  System.out.println("sql 에러 입니다.");  e.printStackTrace();  }  }  }  public static void main(String[] args) {  Main me = new Main();  Student stu = new Student();  Scanner sc = new Scanner(System.in);  System.out.println("학생성적관리 프로그램(ver 0.9.0)");  String input = null;  while (true) {  System.out.print("1.입력 2.보기 3.수정 4.삭제 0.종료 > ");  input = sc.nextLine();  if (input.equals("0")) {  System.out.println("이용해주셔서 감사합니다.");  break;  }  // 입력  if (input.equals("1")) {  System.out.print("학번> ");  input = sc.nextLine();  try {  stu.num = Integer.parseInt(input);  } catch (NumberFormatException e) {  System.out.println("숫자를 입력해주세요");  continue;  }  System.out.print("이름> ");  input = sc.nextLine();  stu.name = input;  System.out.print("국어> ");  input = sc.nextLine();  try {  stu.kor = Integer.parseInt(input);  } catch (NumberFormatException e) {  System.out.println("숫자를 입력해주세요");  continue;  }  System.out.print("영어> ");  input = sc.nextLine();  try {  stu.eng = Integer.parseInt(input);  } catch (NumberFormatException e) {  System.out.println("숫자를 입력해주세요");  continue;  }  System.out.print("수학> ");  input = sc.nextLine();  try {  stu.math = Integer.parseInt(input);  } catch (NumberFormatException e) {  System.out.println("숫자를 입력해주세요");  continue;  }  String sql = "insert into student values (" + stu.num + ",'" + stu.name + "'," + stu.kor + "," + stu.eng + "," + stu.math + ")";  me.write(sql);  }  // 출력  else if (input.equals("2")) {  System.out.print("1. 전체 2. 검색> ");  input = sc.nextLine();  if (input.equals("1")) {  String sql = "select \* from student";  me.print(sql);  } else if (input.equals("2")) {  System.out.print("학번> ");  input = sc.nextLine();  stu.num = Integer.parseInt(input);  String sql = "select \* from student where (num = " + stu.num + ")";  me.print(sql);  } else {  System.out.println("번호를 제대로 입력해주세요");  continue;  }  }  // 수정  else if (input.equals("3")) {  System.out.print("학번> ");  input = sc.nextLine();  try {  stu.num = Integer.parseInt(input);  } catch (NumberFormatException e) {  System.out.println("숫자를 입력해주세요");  continue;  }  System.out.print("국어> ");  input = sc.nextLine();  try {  stu.kor = Integer.parseInt(input);  } catch (NumberFormatException e) {  System.out.println("숫자를 입력해주세요");  continue;  }  System.out.print("영어> ");  input = sc.nextLine();  try {  stu.eng = Integer.parseInt(input);  } catch (NumberFormatException e) {  System.out.println("숫자를 입력해주세요");  continue;  }  System.out.print("수학> ");  input = sc.nextLine();  try {  stu.math = Integer.parseInt(input);  } catch (NumberFormatException e) {  System.out.println("숫자를 입력해주세요");  continue;  }  String sql = "update student set kor='" + stu.kor + "',eng='" + stu.eng + "',math='" + stu.math + "' where (num = " + stu.num + ")";  me.modify(sql);  }  // 삭제  else if (input.equals("4")) {  System.out.print("1. 삭제 2. 전체삭제> ");  input = sc.nextLine();  if (input.equals("1")) {  System.out.print("학번> ");  input = sc.nextLine();  stu.num = Integer.parseInt(input);  String sql = "delete from student where (num = " + stu.num + ")";  me.delete(sql);  System.out.println("삭제되었습니다.");  } else if (input.equals("2")) {  String sql = "delete from student";  me.delete(sql);  System.out.println("전체 삭제 되었습니다.");  } else {  System.out.println("번호를 제대로 입력해주세요");  continue;  }  } else {  System.out.println("번호를 제대로 입력해주세요");  continue;  }  }  }  } | | | |