|  |  |
| --- | --- |
|  | **DEVELOPER JAVA FINAL TEST** |
| **Mã: JAVA01**  **Tổng số câu hỏi: 40**  **Thời gian: 1h30’** |

# Chú ý:

* Chọn 1 hoặc nhiều câu trả lời cho mỗi câu hỏi.
* Được phép sử dụng điện thoại và tài liệu tham khảo trong khi làm bài.

# Câu hỏi và đáp án

|  |  |
| --- | --- |
| **1. Cách để bỏ qua một method trong debug Eclipse là gì?**   1. Sử dụng **step into** 2. Sử dụng **step over** 3. Sử dụng **step return** | |
| **2. Cách tạm dừng chương trình tại 1 câu lệnh trong Eclipse?**   1. Sử dụng break point function 2. Sử dụng pause statement function 3. Sử dụng pause line function | | | |
| **3. Câu lệnh nào đúng?**  byte b = 10;  b = b + 10; // Statement 1  b += 10; // Statement 2   1. Statement 1 và 2 đều compile thành công 2. Statement 1 compile lỗi và statement 2 compile thành công 3. Statement 2 is error compile and statement 1 is OK 4. Statement 1 and 2 are the same runtime exception error. | | | |
| **4.** **output của chương trình sau là gì?**  class Conditional {  public static void main(String args[]) {  int i = 20;  int j = 55;  int z = 0;  z = i < j ? j : i; // ternary operator  System.out.println("The value assigned is " + z);  }  }   1. The value assigned is 20 2. The value assigned is 55 3. The value assigned is 2 4. The value assigned is 0 | | | |
| **5. Chọn đáp án đúng?**  class TestClass {  int x;  public static void main(String[] args){  // TODO.  }  }   1. Khai báo x là static, có thể truy cập được this.x trong method main 2. Khai báo x là public, có thể truy cập được this.x trong method main 3. Khai báo x là protected, có thể truy cập được this.x trong method main 4. Method main không thể truy cập được this.x với bất kỳ access modifier nào | | | |
| **6. cho 1 class có tên Test, lựa chọn nào là định nghĩa hợp lệ cho constructor của class?**   1. Test(Test b) { } 2. Test Test( ) { } 3. private final Test( ) { } 4. void Test( ) { } | | | |
| **7. chức năng của toán tử dot(.) là gì?**   1. Nó cho phép bạn truy cập các instance của bất kỳ đối tượng nào trong một class 2. Nó được sử dụng để gọi phương thức của object 3. Tất cả các điều trên   **8. output của câu lệnh sau là gì?**  System.out.println("// Looks like a comment.");   1. // Looks like a comment 2. compile lỗi 3. Looks like a comment 4. Không có output hiển thị | | | |
| **9. ký tự nào được cho phép là ký tự đầu tiên của 1 định danh?**   1. A – Z, a – z, \_ ,$ 2. A – Z, a – z 3. A – Z, a – z, 0 - 9 4. A – Z, a – z, 0 - 9, \_ ,$ | |
| **10. Có thể loại bỏ tham chiếu (reference) đối tượng bằng cách đặt giá trị của biến có kiểu dữ liệu là kiểu tham chiếu thành \_\_\_?**   1. Nothing 2. zero 3. null 4. empty | |
| **11.** **Object o1 của class A truy cập một property hoặc method của object o2 của class B, khi property hoặc method không có công cụ sửa đổi truy cập, lớp B phải là …**   1. class con của A 2. nằm cùng package với class A 3. class cha của A 4. 1 class con nhưng không ở cùng package với class A | |
| **12. Sử dụng keyword nào để có đa kế thừa trong Java?**   1. Interface 2. Class 3. abstract class 4. object | |
| **13. Chọn đáp án đúng?**  public class Frog {  static int count=0;  public Frog() {  count++;  }  public static void main(String[] args) {  new Frog();  new Frog();  new Frog();  System.out.println("count=" + count);  }  }   1. 1 2. 0 3. 3 | |
| **14.** **Kết quả của việc compile và run đoạn code sau là gì?**  abstract class MineBase {  abstract void amethod();  static int i;  }  public class Mine extends MineBase {  public static void main(String argv[]){  int[] ar=new int[5];  for(i=0;i < ar.length;i++)  System.out.println(ar[i]);  }  }   1. Chuỗi "00000" được in ra 2. Error: Mine could not be instantiiated 3. Error: Mine must be declared abstract 4. IndexOutOfBoundes Error | |
| **15. Interface là gì?**   1. Là 1 class cụ thể 2. Là là 1 superclass 3. Là 1 loại abstract class | |
| **16. Lựa chọn nào đúng?**  public class TestClass {  static int a;  int b;  public TestClass() {  int c;  c = a;  a++;  b += c;  }  public static void main(String args[]) {  new TestClass();  }  }   1. Đoạn code trên sẽ compile lỗi, vì hàm khởi tạo không thể truy cập được thuộc tính static 2. Đoạn code trên sẽ compile lỗi, vì hàm khởi tạo đã sử dụng biến static trước khi biến static được khởi tạo 3. Đoạn code trên sẽ compile lỗi, vì hàm khởi tạo đã sử dụng biến b trước khi biến b được khởi tạo 4. Tất cả các đáp án trên đều không đúng | |
| **17. Kết quả của việc compile và run đoạn code sau là gì?**  public class Law{  public static void main(String [] args){  boolean [] law = {false, true, true};  Object [] obj1 = new Object[1];  String test = “test”;  obj1[0] = test;  System.out.println(law[3]);  }  }   1. Compile lối ở dòng 3 2. Compile lối ở dòng 6 3. Compile không bị lỗi nhưng có 1 exception xảy ra trong khi program thực thi 4. Compile lối ở dòng 8 | |
| **18. Khi không có file "JavaCodingConvention.txt" thì kết quả của việc compile và run đoạn code sau là gì?**  import java.io.\*;  public class Mine {  public static void main (String args[]) {  Mine m = new Mine();  System.out.println(m.amethod());  }  public int amethod() {  try {  FileInputStream dis = new FileInputStream(“JavaCodingConvetion.txt”);  } catch (FileNotFoundException fne) {  System.out.println(“No such file found”);  return -1;  } catch (IOException ioe) {  }  finally {  System.out.println(“Doing finally”);  }  return 0;  }  }   1. No such file found 2. No such file found, -1 3. No such file found, Doing finally, -1 4. 0 | |
| **19. Để sắp xếp một ArrayList <T> theo Class Array, Class T phải được implement method nào?**   1. Comparable<T>.compare. 2. Comparable<T>.compareTo 3. Comparator<T>.compare. 4. Sortable<T>.sortOrder | |
| **20. Khối nào bắt buộc phải thực thi cho dù Exception có xảy ra hay không?**   1. Try 2. Finally 3. Catch | |
| **21. Đoạn code nào, được viết vào TODO mà không bị lỗi khi compile?**  class Nav{  public enum Direction { NORTH, SOUTH, EAST, WEST }  }  public class Sprite{  // TODO  }   1. Direction d = NORTH; 2. Nav.Direction d = NORTH; 3. Direction d = Direction.NORTH; 4. Nav.Direction d = Nav.Direction.NORTH; | |
| **22. Câu lệnh nào sau đây sẽ khởi tạo 1 Inner class?**  public class OuterClass {  public static class InnerClass {  public static void foo() {  // TODO  }  }  }   1. OuterClass.InnerClass m = new OuterClass.InnerClass(); 2. OuterClass.InnerClass mi = new InnerClass(); 3. OuterClass m = new OuterClass();   OuterClass.InnerClass mi = m.new OuterClass.InnerClass();   1. InnerClass mi = new OuterClass.InnerClass(); | |
| **23. câu lệnh nào sau đây sẽ khởi tạo vào InnerClass ở TODO?**  class OuterClass {  class InnerClass{  }  }  class InnerClassTest {  public static void main (String [] args) {  OuterClass f = new OuterClass();  /\* TODO \*/  }  }   1. OuterClass.InnerClass b = new OuterClass.InnerClass(); 2. OuterClass.InnerClass b = f.new InnerClass(); 3. InnerClass b = new f.InnerClass(); 4. InnerClass b = f.new InnerClass(); | |
| **24. \_\_\_\_\_\_\_ giúp quản lý kết nối giữa chương trình Java và cơ sở dữ liệu?**   1. ResultSet 2. Statement 3. Connection 4. java.sql | | | |
| **25.** **Một đối tượng \_\_\_\_\_\_\_\_\_\_ chứa thông tin về tập kết quả như số và tên của các column trong tập kết quả**   1. Prepared statements 2. ResultSet 3. Connection pooling 4. OR mapping (object to relational) | | | |
| **26.** **1 file lưu trữ một hoặc nhiều câu lệnh SQL được gọi là \_\_\_\_\_\_\_\_\_**   1. SQL script 2. SQL statement 3. Result set 4. Connection | | | |
| **27.** **Thứ tự các bước để kết nối JDBC với database?**  1) Tạo kết nối JDBC tới Database.  2) Lấy kết quả của câu query trả ra.  3) Thực thi câu query.  4) Đóng kết nối với database   1. 1-2-3-4 2. 4-3-2-1 3. 1-3-2-4 4. 2-3-4-1 | | | |
| **28. Một method lấy một object Student và ghi nó vào bảng Sinh viên trong cơ sở dữ liệu là \_\_\_\_\_\_\_\_\_\_\_?**   1. OR mapping (object to relational) 2. init, destroy 3. ResultSet 4. Connection pooling | | | |
| **29. Có bao nhiêu kiểu dữ liệu nguyên thủy (primitive types) trong Java?**   1. 7: int, long, float, short, char, boolean, byte 2. 9: int, long, float, short, char, boolean, byte, string, double 3. 8: int, long, float, short, char, boolean, byte, double 4. Không có câu trả lời nào đúng. | | | |
| **30.** **Đâu là lựa chọn đúng về JDBC drivers?**   1. Native-API/partly Java driver (Native) 2. All Java/Net-protocol driver (Middleware) 3. All Java/Native-protocol driver (Pure) 4. JDBC-ODBC Bridge driver (Bridge) | | | |
| **31.** **Ký tự nào được sử dụng để thể hiện một tham số đầu vào trong câu SQL ở prepared statement?**   1. % 2. \* 3. ? 4. # | | | |
| **32.** **Interface nào của API JDBC hứng kết quả của câu lệnh SQL?**   1. PreparedStatement 2. Statement 3. Resultset | | | |
| **33.** **Class thuộc java.sql nào cung cấp method getConnection()?**   1. Connection 2. ResultSet 3. Driver 4. DriverManager | | | |
| **34. Method nào được sử dụng để thực hiện câu lệnh DML trong JDBC?**   1. execute() 2. executeQuery() 3. executeUpdate() 4. executeResult() | | | |
| **35.** **Câu lệnh nào sau đây sẽ xóa tất cả các records khỏi table Invoices có InvoiceTotal = 0?**  String query = "DELETE FROM Invoices " + "WHERE InvoiceTotal = 0 ";  Statement statement = connection.createStatement();   1. statement.executeQuery(query); 2. statement.executeUpdate(query); 3. statement.deleteRows(query); 4. statement.delete(query); | | | |
| **36.** **Điều nào sau đây sẽ nhận được data từ cột đầu tiên của ResultSet rs, được trả về từ việc thực hiện câu lệnh SQL sau:**  SELECT name, rank, serialNo FROM employee;   1. rs.getString(0) 2. rs.getString(1) 3. rs.getString(“rank”) | | | |
| **37.** **Bạn cần một \_\_\_\_ để truy cập database từ Java.?**   1. JDBC 2. Servlet 3. Connection 4. ResultSet | | | |
| **38.** **1 constructor tự động được gọi khi 1 object được khởi tạo có đúng ko?**   1. TRUE 2. FALSE | | | |
| **39.** **Khi có instance của một đối tượng thì tất cả các đối tượng có thể truy cập được từ instance được lưu trữ không?**   1. FALSE 2. TRUE | | | |
| **40.** **do method finalize() thuộc về the java.lang.Object class, nên nó sẽ có mặt trong tất cả các \_\_\_**   1. objects 2. classes 3. methods | | | |