# Java Bronze　後期中間試験　令和1年12月10日　実施

## カプセル化の説明として、正しいものを一つ選びなさい。 （　C　）

1. 同種の異なるインスタンスを同じ型として扱う
2. 公開すべきものと非公開にすうべきものを区別して扱う
3. 関係するものをひとまとめにする
4. クラスのフィールドを隠蔽し、他のクラスからは直接使えないようにする。

## 従業員の情報を表すEmployeeくらすがある。正しくカプセル化されるようにこのクラスを修正したい。修正内容として、正しいものを2つ選びなさい。 （　C,D　）

|  |
| --- |
| public class Employee{  String corporateName;  String corporateAddress;  int employeeNo;  String name;  } |

1. すべてのフィールドのアクセス修飾子をprivateにする
2. フィールドにアクセするためのgetter、setterメソッドを追加する
3. corporateNameとcorporateAddressフィールドを他のクラスに移動する
4. 名前を名乗ってあいさつするメソッドを追加する
5. 給与計算のメソッドを追加する

## カプセル化の維持に欠かせない原則として、正しいものを1つ選びなさい。 （　A　）

1. データ隠蔽
2. 情報隠蔽
3. 抽象化
4. データ抽象

## データ隠蔽を実現するためには、フィールドをどのように修飾すればよいか。正しいものを1つ選びなさい。 （　C　）

1. public final
2. public static
3. private
4. private static
5. private final

## データ隠蔽の説明として、最も適切なものを1つえらびなさい。 （　B　）

1. 関係するデータとそのデータを扱う処理をひとまとめにする
2. 属性の公開範囲を制限する
3. getterやsetterなどのアクセサメソッドを提供する
4. 実装の詳細を隠蔽する

## 共通部分だけを抽出し、それ以外を無視して扱うことを何と呼ぶか。正しい用語を選びなさい。 （　A　）

1. 抽象化
2. カプセル化
3. データ隠蔽
4. 情報隠蔽

## 次のプログラムをコンパイル、実行した時の結果として、正しいものを選びなさい。 （　B　）

|  |
| --- |
| public class A{  public void test(){  System.out.println( "A");  }  } |

|  |
| --- |
| public class B{  public void test(){  System.out.println( "B");  }  } |

|  |
| --- |
| public class Main{  public static void main( String [ ] args ){  A a = new B();  a.test();  }  } |

## クラス定義の記述として、正しいものを１つ選びなさい。 （　　）

1. class ClassA ( int a ){  
    // クラス内定義  
   }
2. ClassA {  
    // クラス内定義  
   }
3. class ClassA ( ) {  
    // クラス内定義  
   }
4. class ClassA {  
    // クラス内定義  
   }