**每日作业卷**

**Java基础第15天**

**反射**

传智播客.黑马程序员

# 关卡1

## 训练案例1

### 训练描述：【讲义：1.1—类的加载】

1. 叙述类加载的过程

**答案：参见讲义1.1 类的加载**

### 操作步骤描述

（参考讲义1.1）

## 训练案例7

### 训练描述：【讲义1.1—类的加载】

4下列哪个选项不属于Java程序加载类的步骤( )   
  
A、加载：读取class文件  
B、连接：验证内部结构，为静态资源分配空间，处理非静态引用  
C、初始化：将代码放到代码区，初始化静态成员，将静态和非静态分离  
D、创建对象：为该类创建一个普通的对象

**答案：D**

### 操作步骤描述

（参考讲义1.1）

## 训练案例3

## **2.1训练描述：【讲义1.2—类的初始化】**

1. 简述常见的四种类的初始化。

**答案：参见讲义1.2类的初始化**

### 2.2操作步骤描述

（参考讲义1.1）

## 训练案例4

### 训练描述：【讲义1.5—知识储备】

1. 说下一下几个单词的含义
   * Class
   * Constructor
   * Method
   * Field
   * instance
   * invoke

.

**答案：参见讲义1.5-知识储备**

### 操作步骤描述

（参考讲义1.5）

## 训练案例4

### 训练描述：【讲义1.6—Class获得方式】

1. 说一下获取Class字节码文件对象的三种方式

**答案：参见讲义1.6-Class获得方式**

### 操作步骤描述

（参考讲义1.6）

## 训练案例6

### 训练描述：【讲义1.6—Class获得方式】

一、已知 一个类的路径 是cn.itcast.demo.User

根据该路径 获取到 User类的字节码文件对象

**答案：**

**day15作业\_Test1\_06**

### 操作步骤描述

（参考讲义1.6）

## 训练案例6

### 训练描述：【讲义1.7—构造方法与实例】

1. 创建一个包 cn.itcast.demo7 写一个User类 并且提供空参构造

写一个MainApp类 要求根据提供的信息 创建一个User对象,不能使用new的方式

也就是要采用反射的方式创建对象

**答案：day15作业\_Test1\_07**

### 操作步骤描述

（参考讲义1.7.2）

## 训练案例8

### 训练描述：【讲义1.8—方法与执行】

一、编写一个类，增加一个实例方法用于打印一条字符串。

并使用反射手段创建该类的对象， 并调用该对象中的方法

**答案：day15作业\_Test1\_08**

### 操作步骤描述

1：创建一个类 写出空参构造与一个实例方法

2：反射方式获取对象

3：反射方式获取那个Method对象

4: 通过invoke方法执行

## 训练案例9

### 训练描述：【讲义1.8—方法与执行】

一、假定Tester 类有如下test方法：  
public int test(int p1, Integer p2)  
以下哪段代码能正确地动态调用一个Tester 对象的test方法？（单选）  
A．  
Class classType=Tester.class;  
Object tester=classType.newInstance();  
Method addMethod=classType.getMethod("test",new Class[]{int.class,int.class});  
Object result=addMethod.invoke(tester,  
new Object[]{new Integer(100),new Integer(200)});  
B．  
Class classType=Tester.class;  
Object tester=classType.newInstance();  
Method addMethod=classType.getMethod("test",new Class[]{int.class,int.class});  
int result=addMethod.invoke(tester,  
new Object[]{new Integer(100),new Integer(200)});

C．  
Class classType=Tester.class;  
Object tester=classType.newInstance();  
Method addMethod=classType.getMethod("test",new Class[]{int.class,Integer.class});  
Object result=addMethod.invoke(tester,  
new Object[]{new Integer(100),new Integer(200)});  
D．  
Class classType=Tester.class;  
Object tester=classType.newInstance();  
Method addMethod=classType.getMethod("test",new Class[]{int.class,Integer.class});  
Integer result=addMethod.invoke(tester,  
new Object[]{new Integer(100),new Integer(200)});

**答案：C**

### 操作步骤描述

（参考讲义1.8）

## 训练案例10

### 训练描述：【讲义--1.9字段(成员变量)与数据操作】

一、使用反射机制获取一个类的属性，下列关于getField()方法说法正确的是( )   
A、该方法需要一个String类型的参数来指定要获取的属性名  
B、该方法只能获取私有属性  
C、该方法只能获取公有属性  
D、该方法可以获取私有属性，但使用前必须先调用setAccessible(true)

**答案：C**

### 操作步骤描述

（参考讲义1.9）

# 关卡2

## 训练案例1

### 训练描述：【反射方法综合练习】

一、编写一个类A，增加一个实例方法showString,用于打印一条字符串，

在编写一个类TestA ,作为客户端，用键盘输入一个字符串，该字符串就是类A的全名，使用反射机制创建该类的对象，

并调用该对象中的方法showString

**答案：见附件项目目录：day15作业\_Test2\_1**

### 操作步骤描述

**见附件项目目录：day15作业\_Test2\_1**

## 训练案例2

### 训练描述：【反射方法综合练习】

一、定义一个标准的Person类，包含属性name，包含setget方法,包含空参构造。

使用反射的方式创建一个实例、并初始化，使用反射方式调用setName方法对名称进行设置,设置之后 在通过反射方式执行getName()方法

**答案：见附件项目目录：day15作业\_Test2\_2**

### 操作步骤描述

1. 建立项目：day15作业\_Test2\_2
2. 创建一个Person类 包含空参构造,name属性,setget方法
3. 创建MiainApp类，包含main()方法
4. 通过反射方式获取字节码文件对象
5. 快捷方式创建对象
6. 反射获取setName方法并设置一个name属性值
7. 反射方式 执行getName 并将返回值打印

# 关卡3

## 训练案例1

### 训练描述：【反射综合案例】

一、ArrayList<Integer> list = new ArrayList<Integer>();

这个泛型为Integer的ArrayList中存放一个String类型的对象

**答案：见附件项目目录：day15作业\_Test3\_1**

### 操作步骤描述

1：创建集合对象

2：添加数字可以 但是添加字符串就报错 这时候想到反射

3：通过对象获得字节码文件对象

4：通过字节码文件对象得到add方法反射形式

5: 反射方式 执行add 添加字符串

## 训练案例2

### 训练描述：【反射综合案例】

一、已知一个类，定义如下：

package com.itcast.demo;

public class DemoClass {

public void run() {

System.out.println("welcome to heima!");

}

}

(1)写一个Properties格式的配置文件，配置类的完整名称。

(2) 写一个程序，读取这个Properties配置文件，获得类的完整名称并加载这个类，

(3)用反射 的方式运行run方法。

**答案：见附件项目目录：day15作业\_Test3\_1**

### 操作步骤描述

1：按要求定义一个类

2：写一个配置文件 键值对形势 className = 值为类的路径

3：创建一个MainApp类 定义main方法

4：在main方法中创建一个Properties集合类

5: 调用读取功能将文件中的路径读取出来 获取字节码对象

6: 反射获取对象

7: 反射获取run方法 并执行