

第一题:

1. **Class**
类。所有属于某一类对象的变量和方法的集合。
2. **abstract class**
抽象类。含有虚函数的类，不能实例化。
3. **virtual function**
虚函数。实现多态性的成员函数。
4. **pure virtual function**
纯虚函数。不需要定义的虚函数。

第二题

```
Class 动物{
public:
virtual void run()=0;
virtual void eat()=0;
virtual bool hair()=0;
virtual string color()=0;
}
```

```
Class 猫科:public 动物{
public:
virtual void run()=0;
virtual void eat()=0;
virtual bool hair()=0;
virtual string color()=0;
virtual string place()=0;
virtual string size()=0;
}
```

```
Class 猫: public 猫科{
public:
virtual void run()=0;
virtual void eat()=0;
virtual bool hair()=0;
virtual string color()=0;
virtual string place()=0;
virtual string size()=0;
}
```

```
Class 波斯猫: public 猫{
public:
virtual void run()=0;
virtual void eat()=0;
virtual bool hair()=0;
virtual string color()=0;
```

```
virtual string place()=0;
virtual string size()=0;
}
```

```
Class 老虎: public 猫科{
public:
virtual void run()=0;
virtual void eat()=0;
virtual bool hair()=0;
virtual string color()=0;
virtual string place()=0;
virtual string size()=0;
}
```

```
Class 东北虎: public 老虎{
public:
virtual void run()=0;
virtual void eat()=0;
virtual bool hair()=0;
virtual string color()=0;
virtual string size()=0;
}
```

第三题

Java

类 (class)

类似于 C++ 的 class

Java 的类只能有一个超类，但可以有多个接口

C++ 的类可以多重继承

方法(method)

类似于 C++ 的 virtual function，可以重载。

接口(interface)

类似于 C++ 的 abstract class

接口中所有的方法都是 abstract 的，而 abstract class 可以有非 abstract 的方法

重写(override)

类似于 C++ 中的虚函数继承及重载。