

Conceptos fundamentales de Java 4-2: Clases de objeto y piloto Actividades prácticas

# Objetivos de la lección:

* Describir la forma general de un programa Java
* Describir la diferencia entre una clase de objeto y una clase de controlador
* Acceder a un mínimo de dos API de clase Java
* Explicar y dar ejemplos de palabras clave de Java
* Crear clases de objeto
* Crear clases de controlador

# Vocabulario

Identifique el término correspondiente a cada una de las definiciones a continuación.

|  |  |
| --- | --- |
| Proyecto | Un grupo de clases Java relacionadas. |
| Clases y condicionales | Secciones de código que se escriben entre llaves. {} |
| Clase con formato CamelCase | Primera letra mayúscula y mayúscula inicial en la primera letra de cada palabra interna. Ejemplo: CuentaCorriente |
| Constante o static final | Valor con nombre que no cambia. |
| Variable con formato CamelCase | Primera letra minúscula y mayúscula inicial en la primera letra de cada palabra interna. Ejemplo: nombreEstudiante |
| Main | Clase que contiene un método principal. |
| Import | Sentencia de código en un archivo de clase Java que incluye código Java de otro paquete o clase. |
| Clase variable = new clase (); | Clase que define las instancias de objetos que hay que usar en otra clase. |
| //Los comentarios se pueden usar en cualquier parte del código | Código precedido por //. Los comentarios se utilizan para aclarar la lógica de programación. El compilador ignora los comentarios. |
| Palabras reservadas como tipos de datos | Palabra que tiene una función especial en el lenguaje Java y no puede utilizarse como nombre de clase, método o variable. |
| Paquete | Biblioteca de clases Java disponible para importar en una clase creada por el programador. |
| TipoDeDato NombreVariable; | Estructura de un objeto, incluidas las variables de clase, los constructores y los métodos. |
| Clase variable = new clase (); | Tipo especial de método que es una plantilla para un objeto. |
| Getter | Valores que se envían a un método o constructor para que se utilicen en un cálculo o para sustituirlos por valores de la clase. |
| Son definidos con un int | Valores como números, caracteres o booleanos. Referencias a objetos como un objeto CuentaBancaria. |
| Public, protected, private, static, final | Palabras clave para especificar la accesibilidad de una clase (o tipo) y sus miembros. Por ejemplo: public, private, protected, default |
| Setter and getter | Bloque de código dentro de una clase que se usa para cambiar la información de la clase o acceder a ella. |

# Inténtelo/resuélvalo:

1. Enumere los componentes que forman un archivo .java. Enumere los componentes en el orden en que esperaría verlos en un programa Java.

R//

1. Nombre de la clase

2. Metodos

3. Constructores

1. Describa la diferencia entre el formato CamelCase en mayúsculas y en minúsculas y proporcione un ejemplo de cuándo los usaría.

R//

En minúsculas se pondría la primer mayúscula tras unión de palabras y en mayúscula desde la inicial

Ej: Mayúscula: TablaDeRespuestas

Minúscula: tablaDeRespuestas

1. ¿Qué sintaxis se utiliza para importar todo el paquete de utilidades de Java? Y si importase todo el paquete, ¿también necesitaría importar las clases adicionales en el mismo paquete por separado?

R//

import NomPaq.Clase.NomVar

Si, lo adecuado es importar una por una las clases adicionales, aunque se puede importar el paquete completo

1. Escriba la sintaxis de una clase de objeto Java simple con el nombre Student y el siguiente formato: Student Name: Lisa Palombo

Student ID: 123456789 Student Status: Active

La información del estudiante se almacenará en las siguientes variables:

fName, lName, stuId, stuStatus.

R//

package ejeruno;

public class Student {

String fName="Lisa";

String lName="Palombo";

double stuId =123456789;

String stuStatus="Active";

}

1. Escriba el código para una clase de controlador que cree un objeto Student y muestre la información sobre el objeto en la pantalla.

R//

package ejeruno;

public class Principal {

public static void main(String[] args) {

Student std = new Student();

System.out.println(std.stuID);

}

}

1. Enumere 10 palabras clave de Java de esta lección.

R//

1. Clase

2. paquete

3. importar

4. class

5. Constructores

6. Métodos

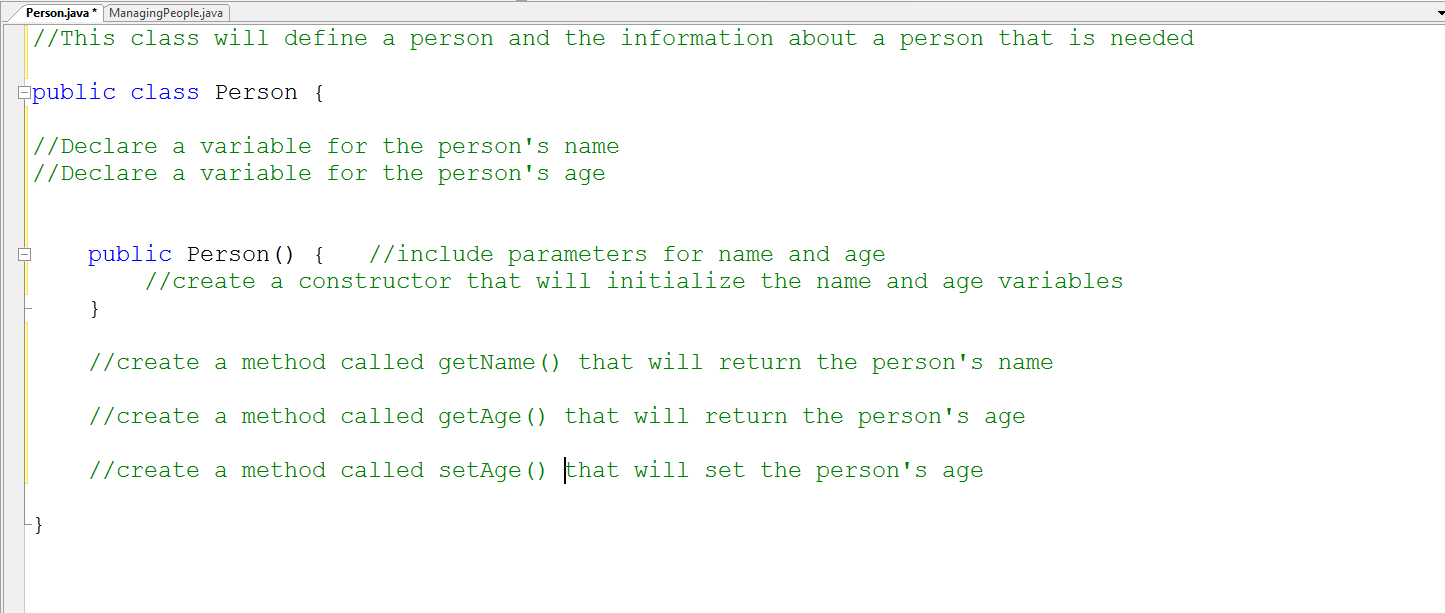
7. String

8. int

9. public

10. double

1. Complete la clase de objeto creada por el programador que aparece a continuación. Tenga en cuenta las instrucciones de los comentarios.



R//

public class Person {

private String name;

private int age;

public Person(String name, int age) {

super();

this.name = name;

this.age = age;

}

public String getName() {

return name;

}

public int getAge() {

e

rn age;

}

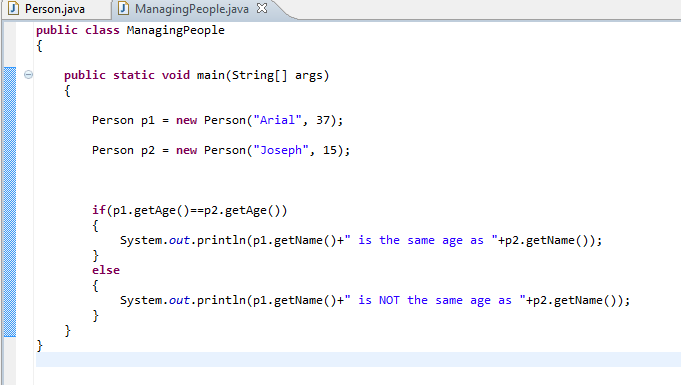
public void setAge(int age) {

this.age = age;

}

}

1. Complete la siguiente clase de controlador. Tenga en cuenta las instrucciones de los comentarios.



R//

public class ManagingPeople

{

Public static void main(String[] args)

{

Person p1 = new Person(“Arial”, 37);

Person p2 = new Person(“Joseph”, 15);

if(p1.getAge()==p2.getAge())

{

System.out.println(p1.getName()+” tiene la misma edad que ”+p2.getName());

}

else

{

System.out.println(p1.getName()+” no tiene la misma edad que ”+p2.getName());

}