



Carátula Para Entrega De Prácticas

Facultad de Ingeniería

Laboratorios de docencia

Laboratorio de Computación Salas A y B

Profesor(a): José Antonio Ayala Barbosa

Asignatura: Programación Orientada a Objetos

Grupo: 8

No de Práctica(s): Practica #1 “Entorno y Lenguaje De Programación”

Integrante(s): Galindo Mayer Johann Zair

*No. de Equipo de
cómputo empleado:*

Semestre: 2024 - 1

Fecha de entrega: 27/08/2023

Observaciones:

CALIFICACIÓN:

Practica 1 “Entorno y Lenguaje De Programación”

Cuestionario Previo a La Practica

1. ¿Qué es una IDE? (Entorno De Desarrollo Integrado)

Es una aplicación de software que ayuda a los programadores a desarrollar códigos de manera eficiente. Aumenta la productividad de los desarrolladores al combinar capacidades como editar, crear, probar y empaquetar software en una aplicación fácil de usar. Así como los escritores utilizan editores de texto y los contables, hojas de cálculo, los desarrolladores de software utilizan IDE para facilitar su trabajo. Su importancia es proporcionar una interfaz central para herramientas de desarrollo comunes, lo que hace que el proceso de desarrollo de software sea mucho más eficiente.

2. ¿Qué es una API?

Son mecanismos que permiten a dos componentes de software comunicarse entre si

mediante un conjunto de definiciones y protocolos. Un ejemplo sería el sistema de software del instituto de meteorología contiene datos meteorológicos diarios. La aplicación meteorológica de los celulares “habla” con este sistema a través de las API y le muestran las actualizaciones meteorológicas diarias.

3. En programación, ¿Qué es una biblioteca?

Es un conjunto de archivos que se utiliza para desarrollar software. Suele estar compuesta de códigos y datos, y su fin es ser utilizada por otros programadores de forma totalmente autónoma. Simple y llanamente, es un archivo importable.

4. ¿Qué es la compilación de un programa?

Es un proceso de transformar un programa informático escrito en un lenguaje en un conjunto de instrucciones en otro formato o lenguaje. Un compilador es un programa de computadora que realiza dicha tarea.

5. ¿Cuáles son los IDEs más importantes? (Al menos 4)

- Eclipse
- SlickEdit
- CODE::BLOCKS
- GNAT Studio
- AWS Cloud9

- CodeLite
- NetBeans
- JetBrains
- Selenium
-

6. ¿Qué es una herramienta de desarrollo (SDK)

(Kit De Desarrollo De Software)

Es un conjunto de herramientas de creación específicas de plataformas para desarrolladores. Se necesitan componentes como depuradores, compiladores y bibliotecas para crear código que se ejecute en una plataforma, sistema operativo o lenguaje de programación específicos. Los SDK ponen todo lo que necesita para desarrollar y ejecutar software en un solo lugar. Además, contienen recursos como documentación, tutoriales y guías, así como API y marcos para un desarrollo de aplicaciones más rápido.

7. ¿Para qué sirven los comentarios en un programa?

Es texto que el compilador omite pero que es útil para los programadores. Los comentarios se usan normalmente para anotar código para su referencia futura. El compilador los trata como si fuera espacios en blanco.

8. ¿Tipos de comentarios en Java?

➤ Comentarios de línea

Se agregan breves anotaciones para dar identificaciones sencillas. Ocupa menos de una fila, por eso se conocen como comentarios de línea.

Para incluirlas dentro del código, lo primero que hay que hacer es abrir la línea con dos barras (//) y luego el comentario.

➤ Comentarios de bloque

Cuando las indicaciones ocupan más de una línea de código en Java se conocen como comentarios de bloque. Además de servir para ofrecer una explicación más extensa, puede emplearse para deshabilitar temporalmente partes del código, por el motivo que sea.

Para iniciar un comentario de bloque, se debe abrir la notación con una barra y asterisco (*). Una vez introduzcas el

texto, podrás cerrar el mensaje con un asterisco y una barra (*).

➤ Comentarios JavaDoc

Estos comentarios no solo sirven para darles indicaciones a otras personas, también pueden darle instrucciones al propio sistema. Aquí es donde entran los comentarios JavaDoc, en los que el desarrollador incluye una serie de etiquetas para generar automáticamente una documentación técnica y legible en HTML a partir del código fuente.

Para introducir un comentario JavaDoc, primero debes de abrir la indicación con una barra y dos asteriscos (/**) y una vez introduzcas el comentario y las etiquetas adecuadas, debes cerrarlo con un asterisco y una barra (*). Las etiquetas se abren con una arroba @.

9. ¿Qué es un identificador?

Los “identificadores” o “símbolos” son los nombres que se proporcionan para variables, tipos, funciones y etiquetas del programa. Los nombres de identificadores deben diferir en ortografía y mayúsculas y minúsculas de cualquier palabra claves.

10. ¿Qué es Sanke Case y Camel Case?

➤ Sanke Case

Combina las palabras usando un guion bajo como nexos. Existen dos versiones de esta notación, una en la que todas las letras están en minúscula y otra en la que todas las letras están en mayúsculas. Esta notación, cuando se usa en mayúscula, es habitual en la declaración de constantes de lenguajes como PHP o JavaScript.

- La sintaxis de contar palabras en notación Snake Case sería contar_palabras.
- La sintaxis de aumentar nivel dificultad en notación Snake Case sería aumentar_nivel_dificultad.

➤ Camel Case

Combina las palabras directamente, sin usar ningún símbolo establecido que la primera letra de cada palabra este en mayúsculas a excepción de la primera palabra, estando el resto de las letras en minúsculas. Este tipo de notaciones esta muy extendida, siendo su uso muy común tanto en la declaración de variables como en el nombre de funciones y métodos.

Ejemplo:

- La sintaxis de contar palabras en notación Camel Case sería contarPalabras.
- La sintaxis de aumentar nivel dificultad en notación Camel Case sería aumentarNivelDificultad.

11. ¿Qué es una palabra reservada?

Las palabras reservadas en programación, o palabras clave, tienen un significado especial para el compilador de cualquier lenguaje de programación.

Estas palabras pueden identificar los tipos de datos que se pueden usar, además de las diferentes rutinas de programación que permite cada lenguaje.

Referencias:

1. Admin. (2019). ¿Qué son las palabras reservadas en programación? *Lenguajes de programación*. <https://lenguajesdeprogramacion.net/diccionario/que-son-las-palabras-reservadas-en-programacion/>
2. Arias, A. (2023). Comentarios en Java: guía completa para mejorar tu código. *Tokio School*. <https://www.tokioschool.com/noticias/comentarios-java/>
3. *Compile - Glosario de MDN Web Docs: Definiciones de términos relacionados con la web* / MDN. (2023, 18 julio). <https://developer.mozilla.org/es/docs/Glossary/Compile>
4. Gómez, P. (2021, 27 octubre). *Qué es una librería en programación* - DevCamp. DevCamp. <https://devcamp.es/que-es-libreria-programacion/>
5. Kumar, C. (2023). 13 mejores IDE que todo programador debe conocer. *Geekflare*. <https://geekflare.com/es/ide-for-programmer/>
6. Lázaro, E. (2023, 20 enero). *Tipos de notación: camel case, pascal case, snake case y kebab case* | Neoguias. Neoguias | Sistemas, Aplicaciones, Dispositivos y Programación a Todos los Niveles. <https://www.neoguias.com/tipos-notacion-nombres/>
7. *¿Qué es el SDK? - Explicación del SDK* - AWS. (s. f.). Amazon Web Services, Inc. <https://aws.amazon.com/es/what-is/sdk/>
8. *¿Qué es un IDE? - Explicación de los entornos de Desarrollo Integrado* - AWS. (s. f.). Amazon Web Services, Inc. <https://aws.amazon.com/es/what-is/ide/>
9. *¿Qué es una API? - Explicación de interfaz de programación de aplicaciones* - AWS. (s. f.). Amazon Web Services, Inc. <https://aws.amazon.com/es/what-is/api/>
10. TylerMSFT. (2023a, abril 3). *Comentarios (C++)*. Microsoft Learn. <https://learn.microsoft.com/es-es/cpp/cpp/comments-cpp?view=msvc-170>
11. TylerMSFT. (2023b, abril 3). *Identificadores de C*. Microsoft Learn. <https://learn.microsoft.com/es-es/cpp/c-language/c-identifiers?view=msvc-170>

Objetivo De La Practica

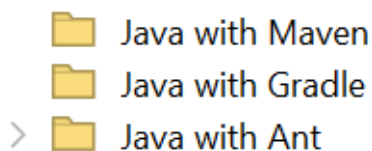
Identificar y probar el entorno de ejecución y lenguaje de programación orientado a objetos a utilizar durante el curso.

Introducción

En la primer practica de laboratorio de la asignatura Programación Orientada a Objetos, se vera la forma de trabajar usando una IDE, se conocerá la sintaxis del lenguaje de programación Java y se escriba el primer Hola Mundo en Java además de recordar conceptos abordados en las clases de Teoría de la asignatura.

Desarrollo De La Practica

Utilizamos la IDE Apache NetBeans para realizar nuestro primer proyecto en Java, y para eso tenemos dos tipos de opciones a la hora de crear nuestro primer proyecto tenemos la opción de usar Java with Maven y Java with Ant.



Java with Maven nos permite utilizar todos los paquetes disponibles en Java ya que se conecta a un servidor en internet y está en constante actualización, pero tiene una desventaja, ya que al correr el programa este tiende a tardar un tiempo significativo. **Java with Ant** nos permite utilizar los paquetes instalados en nuestro ordenador y el tiempo de respuesta al correr el programa es de casi 1 segundo a lo mucho, es decir, es una respuesta instantánea

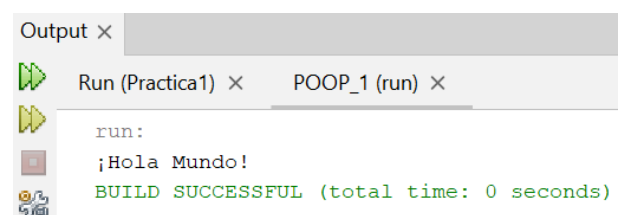
básicamente, pero lo malo de esto es que solo utilizara paquetes instalados, así que si no tenemos los paquetes actualizados puede que perdamos de distintas funciones.

Después de esta explicación procedemos a realizar nuestro primer Hola Mundo en Java, para esto usamos la siguiente sintaxis **System.out.println("Lo que deseamos mostrar en la terminal a la hora de ejecutar el programa");**

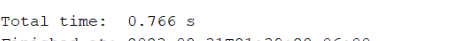
Explicación del System.out.println(" ");

El **system.out** indica que es un objeto que será utilizado en la clase o función donde se localiza.

Print es la acción que realizará, en este caso indicará que deberá imprimir algo que se indica dentro de los paréntesis con sus respectivas comas dobles al inicio y al final.

A screenshot of the NetBeans IDE showing the source code for a file named POOP_1.java. The code is a simple Java program that prints 'Hola Mundo!'. It includes a package declaration 'package poop_1;', a class declaration 'public class POOP_1 {', and a main method 'public static void main(String[] args) {'. Inside the main method, there is a call to 'System.out.println("Hola Mundo!");'. The code is highlighted with a yellow background.A screenshot of the NetBeans IDE's 'Output' window. It shows the results of running the program. The first line is 'run:' followed by the output '¡Hola Mundo!'. Below that, it says 'BUILD SUCCESSFUL (total time: 0 seconds)'. The window has tabs for 'Run (Practica1)' and 'POOP_1 (run)'. There are also icons for running, debugging, and other IDE functions.

Hola Mundo, Usando Java with Maven



Output - Run (JavaAplicacion1) x

```

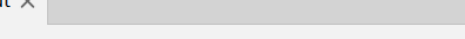
nothing to compile - all classes are up-to-date

--- exec:3.1.0:exec (default-cli) @ JavaAplicacion1 ---
¡Hola Mundo!

-----
BUILD SUCCESS
-----

Total time: 0.766 s
Finished at: 2023-08-31T21:39:28-06:00

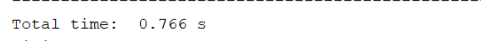
```



Output

Run (Practica1) POOP_1 (run)

```
run:
;Hola Mundo!
BUILD SUCCESSFUL (total time: 0 seconds)
```



```

Output - Run (JavaAplicacion1) X
[Icons] nothing to compile - all classes are up-to-date
--- exec:3.1.0:exec (default-cli) @ JavaAplicacion1 ---
¡Hola Mundo!
-----
BUILD SUCCESS
-----
Total time: 0.766 s
Finished at: 2023-08-31T21:39:28-06:00

```

programar de Java a C, la verdad me confunde un poco.