

Sistemas de Gestión Empresarial – UF2_PAC03

ÍNDICE

1. Define las características principales de Python.	2
2. Características de las variables bajo Python.	2
3. Describe los tipos numéricos básicos.	3
4. Diferencia entre las listas y las tuplas.	3
5. ¿Qué bucle C no existe en Python?	3
6. Cómo se gestionan los errores.	4
7. Tipos de herencia para crear un módulo bajo Open ERP.	4
8. Tipos de vistas existentes en un módulo Open ERP.	4
9. ¿Qué tipo de vistas son graph y search?	5

1. Define las características principales de Python.

- Lenguaje orientado a objetos: Define todo como objetos, implementa la creación de clases y objetos, así como la herencia, pero sin poner límites al acceso de métodos y propiedades que declaremos.
- Lenguaje interpretado multiplataforma: La primera vez que se ejecuta su código fuente se convierte en código objeto.
- Variables: No es necesario declararlas para usarlas, pero sí inicializarlas antes de usarlas. Si no especificamos tipo pueden contener cualquier valor. Para pasar de un tipo a otro requieren de un casting.
- Bloques: No existen marcas para los bloques, sólo se ponen dos puntos (:) para indicar el inicio, el sangrado del código (obligatorio) es el que identifica las sentencias del bloque. El número de espacios del sangrado debe ser igual en todas las líneas para que no de error.
- Fácil de utilizar como script en otros lenguajes: Dispone de versiones en otros lenguajes como C, C# o Java que le hacen fácil de embeber en otros lenguajes.

2. Características de las variables bajo Python.

- No es obligatorio la declaración de la variable para usarla.
- Si es obligatorio inicializarlas antes de usarlas.
- Se puede asignar cualquier valor a la variable, el intérprete será el que determine el tipo de valor almacenado.
- Tenemos dos tipos, variables globales (visibilidad en todo el programa) y locales. (visibilidad en el bloque en el que se haya inicializado). Si se da el mismo nombre a una variable global y otra local al utilizarla realmente se utiliza la local.
- No admite multiasignación en una línea.
- El lenguaje proporciona algunas variables por defecto. Con argv podemos recoger los parámetros de entrada.
- A la hora de imprimir habrá que utilizar los modificadores adecuados según el tipo de dato, convertir a cadena en caso de ser necesario y codificar los caracteres especiales del idioma como la ñ en castellano.

3. Describe los tipos numéricos básicos.

Los tipos numéricos básicos de Python pueden ser de cuatro tipos:

- **Entero simple:** Números enteros positivos o negativos sin punto decimal, la cantidad numérica está delimitada por el sistema en base al tipo de dato.
Ejemplo: 5
- **Entero largo:** Números enteros de tamaño prácticamente ilimitado (restricción en base al sistema), se escriben mediante el entero y seguidos de una "L".
Ejemplo: 12L
- **Coma flotante:** Números reales que utilizan el punto para dividir parte entera y decimal. Si importamos Decimal podremos ampliar el rango. Puede utilizar notación científica.

Ejemplos: 3.5 , 3.2e3 (sería 3.2×10^2)

- **Complejos:** Representa los números complejos, compuestos por una parte real y otra imaginaria, para ello utiliza dos números en coma flotante y la letra "J", la notación sería a+bJ.

Ejemplo: 1.5+3J

4. Diferencia entre las listas y las tuplas.

- Una lista admite modificaciones de los elementos, una tupla no puede alterar ni el número de elementos, no el contenido.
- Una tupla consume menos espacio que una lista.
- Una lista se define mediante corchetes, una tupla mediante paréntesis.
- Las tuplas sólo tiene métodos para acceder por índice a los datos, mientras que las listas además de estos métodos dispone del método para modificar un valor por el índice.

5. ¿Qué bucle C no existe en Python?

El bucle for(;;), pero se puede simular mediante una lista con los números a recorrer y el for in de Python.

6. Cómo se gestionan los errores.

Python dispone de un sistema de excepciones que son lanzadas por el sistema y podemos capturar mediante la estructura `try – except – else – finally`.

El funcionamiento es similar al de otros sistemas, **try** captura las excepciones mediante **except** *excepción_producida*, si no se produce excepciones se ejecuta el bloque **else**; tras el tratamiento de cualquier problema se ejecutará el bloque **finally**.

También podemos crear nuestras excepciones heredando de la clase `Exception` y lanzar excepciones mediante la sentencia **raise**.

Si no controlamos las excepciones el sistema las controlará por defecto parando la ejecución y mostrando un mensaje por consola.

7. Tipos de herencia para crear un módulo bajo Open ERP.

- Herencia por extensión: Añade propiedades a un objeto existente, no crea otro. Se da cuando las propiedades `_name` e `_inherit` son iguales y el objeto asignado existe.
- Herencia por prototipo: Se crea un nuevo objeto con todos los componentes del objeto padre. La propiedad `_name` tendrá un nombre de objeto que no existe e `_inherit` tendrá un objeto que sí existe en el sistema.
- Herencia por delegación: Crea un nuevo objeto heredando las características de varios existentes; herencia múltiple. La propiedad `_name` tendrá un nombre que no existe y la propiedad `_inherits` un diccionario con todos los objetos a heredar.

8. Tipos de vistas existentes en un módulo Open ERP.

- Vista formulario: Se utiliza para la edición de datos:
 - Cada campo tiene una etiqueta con su nombre.
 - Campos de izquierda a derecha y de arriba abajo según el xml.
 - Ventana con 4 columnas y n filas, un campo ocupa dos columnas (etiqueta + campo), pero puede ocupar varias columnas.
 - Se puede dividir una columna en N.
- Vista en árbol: Se usa para los listados de datos.
- Vista de búsqueda: Complementa la vista en árbol, ofreciendo un panel de búsqueda y filtrado.

- Vista de gráfico: Muestra un gráfico a partir de los datos.
- Vista de calendario: Permite mostrar eventos.
- Vistas de Gantt: Permite mostrar gráficos de Gantt.

9. ¿Qué tipo de vistas son graph y search?

Son vistas de gráfico y búsqueda, pero realmente graph es un tipo de vista en árbol y search un complemento a esas vistas en árbol.