
Universidad
Tecmilenio®

Universidad Tecmilenio

Fundamentos de programación para Big Data

Tipos de datos en ambiente de Big Data

Semana 2

Introducción

Las fuentes de donde puedes obtener datos son tan variadas que no puedes limitarte a un tipo de dato en particular. Debes conjuntar los datos desde diferentes fuentes, pero dentro de un contexto único para el cual otorgues información valiosa después de limpiar, organizar y analizar los datos, sin importar los puntos de origen y sin olvidar revisar que las fuentes sean confiables.

Explicación

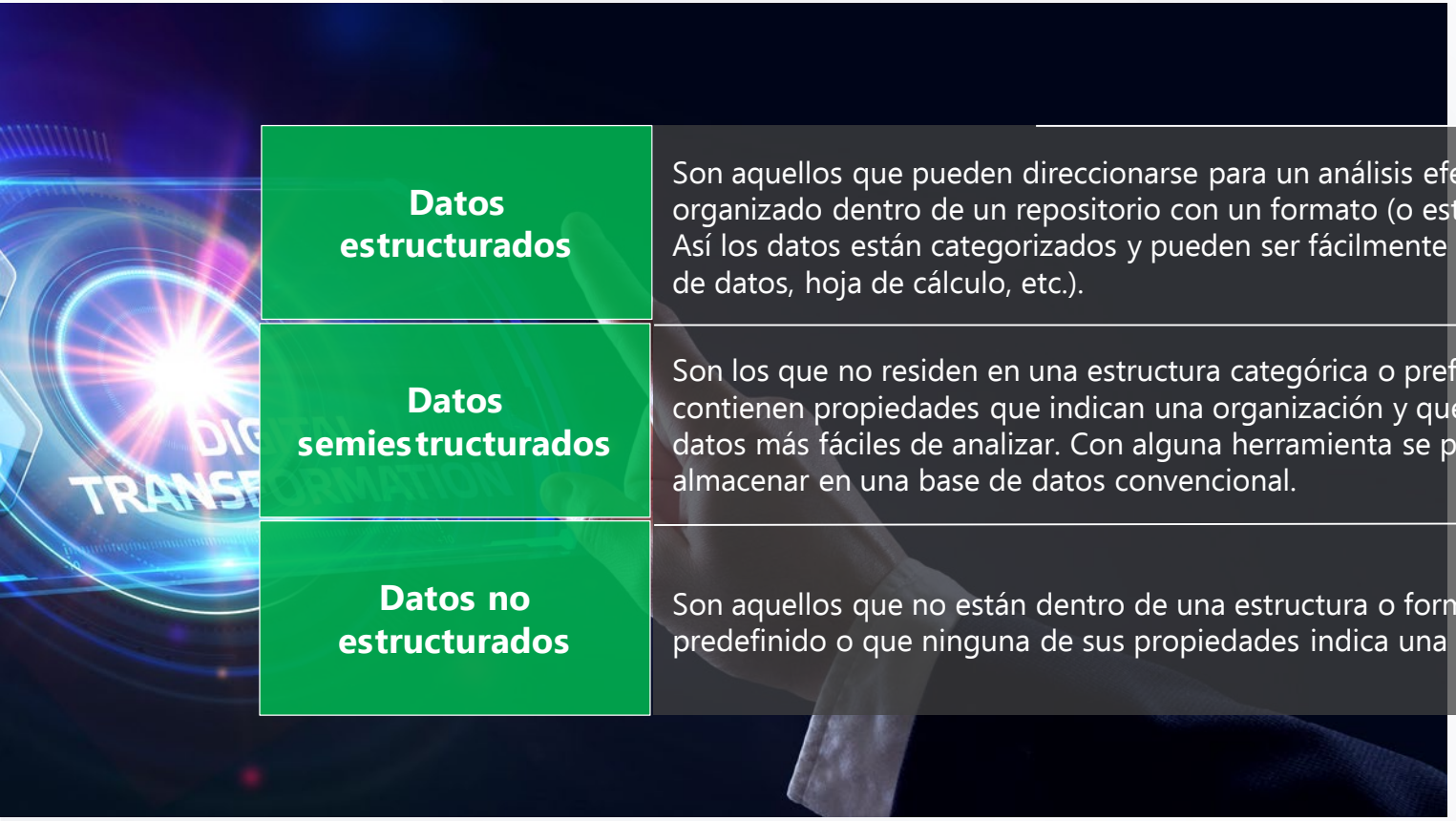
Un **tipo de dato** es la clasificación más básica de un dato. Se transmite entre el programa y el compilador, donde se informa el tipo de dato que será almacenado y cuánto espacio necesita en la memoria para ello (int, string, float, etc.).

Una **estructura de datos** es una colección de estos sobre diferentes tipos de datos. Esta colección organiza sus elementos con respecto a la memoria y a la manera de acceder a cada uno de los elementos para ejecutar una operación definida.

El **uso de los diferentes tipos de datos** en Big Data dependerá del tratamiento necesario que se deba realizar para poder ejecutar el análisis, es decir, se deberán desarrollar las habilidades pertinentes para transformar los datos de un tipo a otro, dependiendo del origen de los datos, pero, sobre todo, el objetivo por el cual estos fueron obtenidos.

Explicación

Los tipos de datos de acuerdo con su estructura pueden clasificarse de la siguiente manera (InteliPaat, 2019):



Datos estructurados	Son aquellos que pueden direccionarse para un análisis efectivo. Estos datos se han organizado dentro de un repositorio con un formato (o estructura) predeterminado. Así los datos están categorizados y pueden ser fácilmente mapeados (tabla de base de datos, hoja de cálculo, etc.).
Datos semiestructurados	Son los que no residen en una estructura categórica o preformateada, pero que sí contienen propiedades que indican una organización y que pueden hacer de estos datos más fáciles de analizar. Con alguna herramienta se pueden manipular y almacenar en una base de datos convencional.
Datos no estructurados	Son aquellos que no están dentro de una estructura o formato contenedor predefinido o que ninguna de sus propiedades indica una organización específica.

Explicación

Las propiedades de los tipos de datos de acuerdo con su estructura son las siguientes (Vishwakarma, 2016):

	Estructurados	Semiestructurados	No estructurados
Tecnología	• Se fundamentan en bases de datos relacionales.	• Basados en XML/RDF.	• Basados en cadenas de caracteres y datos binarios.
Manejo de transacciones	• Transacción madura y varios métodos de concurrencia.	• Transacción no madura.	• Sin administración de transacción y sin métodos de concurrencia.
Manejo de versión	• Versiones sobre tuplas, tablas, etc.	• Versiones sobre tuplas o grafos.	• Versiones de todos los datos como un bloque unificado.
Flexibilidad	• Dependiente del esquema y poco flexible.	• Más flexibles que los estructurados, pero aun dependientes de un esquema.	• Flexible sin esquema.
Escalabilidad	• Escalar vía esquema de BD.	• Escalar vía esquema de BD non sql.	• Es muy escalable.

Cierre

Conocer a profundidad los tipos de datos en Big Data es uno de los pasos fundamentales para usar correctamente cualquier técnica de recolección y análisis de datos.

Así como optimizar el proceso de conversión de un tipo de estructura a otra, según se necesite.

Bibliografía

- IntelliPaat. (2019). *What is Big Data Analytics*. Recuperado de https://intellipaat.com/blog/big- data-analytics/

- Vishwakarma, A. (s.f.). *Difference between Structured, Semi-structured and Unstructured data*. Recuperado de https://www.geeksforgeeks.org/differe nce-between-structured-semi- structured-and-unstructured-data/

Universidad Tecmilenio

Fundamentos de programación para Big Data

Visualización de datos con Big Data

Semana 2

Introducción

Hoy en día es más importante visualizar para la comprensión, debido a que existen múltiples herramientas libres o de licencia que facilitan el manejo de los datos para generar visualizaciones efectivas.

Explicación

La información obtenida de tu análisis puede ser verdadera, pero si no está correctamente representada no será útil de ninguna manera. Debes ser capaz de representarla de una manera sencilla y entendible.

En un segundo nivel, cuando tu información esté más depurada y estructurada, puedes diseñar un *dashboard* o tablero con alguna herramienta colaborativa para poderlo compartir.

Visualizar datos durante el proceso de análisis te ayudará a entender la tendencia de estos; a encontrar patrones, relaciones o datos atípicos para poder determinar si los datos son útiles o no.

Explicación

Es importante considerar lo siguiente al momento de diseñar un tablero o dashboard:

1	Se diseña para la comprensión. Las gráficas deben ser claras e intuitivas.
2	Interactividad siempre que se pueda y que lo amerite.
3	Elige una herramienta que se adapte a tus necesidades, circunstancias y capacidades tecnológicas.

Explicación

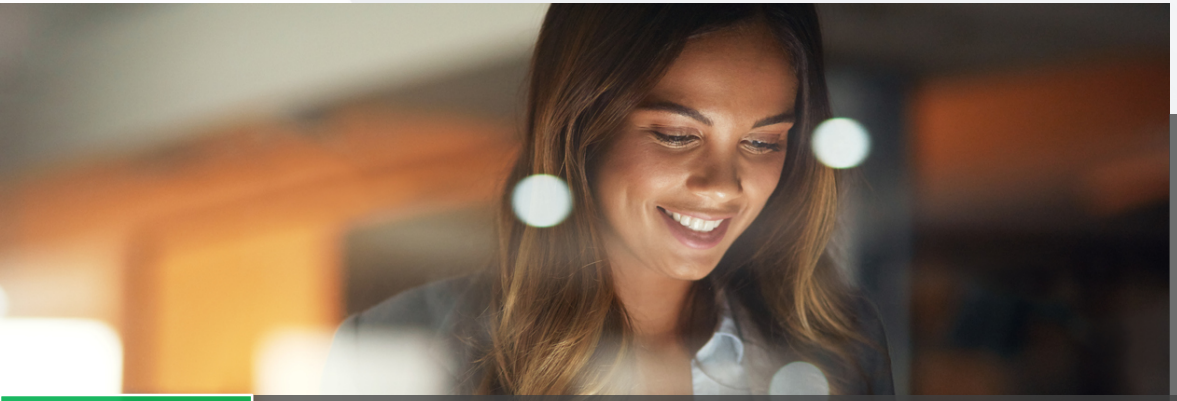


Tableau	La herramienta más popular por excelencia. Además, cubre las ecuaciones, los cálculos más comunes y permite generar fórmulas propias.
Power BI	Herramienta de Microsoft, cuyo motor vio su nacimiento en Excel, por lo tanto, el uso de fórmulas y lenguaje DAX será más que transparente.
Amazon **QuickSight**	Servicio *on cloud*, permite crear y publicar fácilmente paneles interactivos que incluyen información de aprendizaje automático.

Herramientas de licencia

En la actualidad existen herramientas *on desk* o en la nube que te pueden ayudar a hacer un análisis eficiente sin preocuparte de no conocer un lenguaje de programación específico para generar gráficas.

Explicación

Herramientas sin licencia

No estás limitado a una herramienta en particular, tienes muchas opciones. Su uso dependerá de la destreza que tengas y, si tienes bases de programación intermedias, puedes hacer uso de estas herramientas.

Bokeh

Es la librería para Python, la cual te permite generar gráficas de alto desempeño y además exportarlas en HTML para embeberlas en diferentes portales y aplicaciones.

Matplotlib

La librería más usada por analistas y programadores en Python. Se basa en NumPy y Pandas y da versatilidad en el análisis, ya que permite generar la mayoría de las gráficas de manera sencilla, conforme se va realizando el desarrollo.

Grafana

Proyecto de código libre y abierto (en Python) que, si bien, tiene un servicio online público y privado, también cuenta con una versión on desk que puede configurarse a la medida.

Google Cloud, (2016).

Cierre

Una licencia o un producto de moda no es lo que aporta valor a una compañía, sino resolver sus problemas de procesos, reducir sus gastos y no estar obligados a consumir determinada tecnología. Por lo tanto, el mejor producto que puedes ofrecer como programador o persona de tecnología es tu creatividad.

Bibliografía

- Google Cloud. (2016). *Crear paneles de datos interactivos y personalizados con Bokeh y BigQuery*. Recuperado de https://cloud.google.com/solutions/bok eh-and-bigquery-dashboards?hl=es- 419

Universidad Tecmilenio

Fundamentos de programación para Big Data

Programación en lenguaje Python

Semana 2

Introducción

```
});

function checkScroll(scrollPos, introh) {
    if(scrollPos > introh ) {
        header.addClass("fixed");

    } else {
        header.removeClass("fixed");
    }
}
$("[data-scroll]").on("click", function(event) {
    event.preventDefault();
    let elementId = $(this).data('scroll');
    let elementoffset = $(elementId).offset().top;
    nav.removeClass("show");
    $("html, body").animate({
        scrollTop: elementoffset - 70
    }, 700);
});
let nav = $("#nav");
let navToggle = $("#navToggle");

navToggle.on("click", function(event) {
    event.preventDefault();

    nav.toggleClass("show");
```

Actualmente, la informática es una de las áreas con mayor crecimiento y demanda. La programación es convertir una idea dentro de tu cabeza en algo palpable, permitiendo resolver un problema y otorgando una mejor calidad de vida a las personas .

Python, desde el 2017, está dentro de los cinco principales lenguajes de programación más populares (HackerRank, 2019).

Explicación

Python fue creado por Guido Van Rossum en la década de los ochenta, publicando su primera versión en 1991.

Es un lenguaje de programación multiparadigma, es decir, cuenta con soporte para la programación orientada a objetos, programación imperativa y programación funcional.

- Sintaxis elegante.
- Lenguaje sencillo de aprender y de utilizar.
- Gran biblioteca estándar de libre disposición (Pandas y NumPy).
- Software libre.
- Se puede ejecutar en una gran cantidad de plataformas.
- El código se puede agrupar en módulos y paquetes.
- Su tipado es dinámico.
- Cuenta con diversos tipos de datos básicos, como números, cadenas, listas y diccionarios.

Explicación

Para ver el uso de la herramienta Google Colaboratory, te recomendamos ver el siguiente video:

Dirección de Producción de Contenidos. (2019, 4 de diciembre). *Tema 5. Herramienta Google Colaboraty* [Archivo de video]. Recuperado de https://bit.ly/2T02OQN

Para ver una introducción a Python, te recomendamos ver el siguiente video:

Dirección de Producción de Contenidos. (2019, 4 de diciembre). *Tema 5. Introducción a Python* [Archivo de video]. Recuperado de https://bit.ly/2Qx8zUu

Los siguientes enlaces son externos a la Universidad Tecmilenio, al acceder a estos considera que debes apegarte a sus términos y condiciones.

A continuación, se enlistan otros de los editores más populares para usar Python:

- **Jupyter Notebook (Anaconda)**
- **PyCharm**
- **Visual Studio Code**

Explicación

Para saber sobre la sentencia IF en Python, te recomendamos ver el siguiente video:

Dirección de Producción de Contenidos. (2020, 6 de febrero). *Tema 5. Palabra reservada IF.* [Archivo de video]. Recuperado de https://bit.ly/2OvkYXt

Para saber sobre la sentencia WHILE en Python, te recomendamos ver el siguiente video:

Dirección de Producción de Contenidos. (2019, 4 de diciembre). *Tema 5. While en Python* [Archivo de video]. Recuperado de https://bit.ly/2QS2v7X

Para saber sobre las sentencias ELIF y ELSE en Python, te recomendamos ver el siguiente video:

Dirección de Producción de Contenidos. (2019, 4 de diciembre). *Tema 5. ELIF y ELSE en Python* [Archivo de video]. Recuperado de https://bit.ly/36yfnH9

Para saber sobre la sentencia FOR en Python, te recomendamos ver el siguiente video:

Dirección de Producción de Contenidos. (2019, 4 de diciembre). *Tema 5. For en Python* [Archivo de video]. Recuperado de https://bit.ly/36I3c9P

Explicación

Un **módulo** es un archivo (utiliza el sufijo .py) que contiene declaraciones y funciones de Python. Cada módulo incluye su propio *namespace*, el cual se usa como un espacio de nombres global por cada una de las funciones determinadas en el módulo.

Los **paquetes** son una forma de organizar los espacios de nombres en Python. Por ejemplo, el nombre del módulo C.D denomina un submódulo llamado C en el paquete de nombre D.

Para saber sobre la librería Math en Phyton, te recomendamos ver el siguiente video:

Dirección de Producción de Contenidos. (2019, 4 de diciembre). *Tema 5. Librería Math* [Archivo de video]. Recuperado de https://bit.ly/37Q3D2G

Para saber sobre la sentencia WHILE en Python, te recomendamos ver el siguiente video:

Dirección de Producción de Contenidos. (2019, 4 de diciembre). *Tema 5. Funciones en Python* [Archivo de video]. Recuperado de https://bit.ly/2N2UmMY

Dirección de Producción de Contenidos. (2019, 4 de diciembre). *Tema 5. Función range en Python* [Archivo de video]. Recuperado de https://bit.ly/39Jh2eS

Cierre

Python es uno de los lenguajes de programación favoritos de los desarrolladores, gracias a su código limpio y legible, además de su licencia abierta y por ser utilizado en el desarrollo de grandes aplicaciones a nivel mundial, haciéndolo ideal para las personas que desean ingresar a esta industria. Otro punto destacable es que para manejar grandes volúmenes de datos es una excelente opción, en otras palabras, es ideal para Big Data.

Bibliografía

- HackerRank. (2019). *HackerRank Developer Skills Report*. Recuperado de https://research.hackerrank.com/deve loper-skills/2019