



**Universidad autónoma
de Baja California**

**Facultad de Ingeniería,
Arquitectura y Diseño**



Solano Meza Angel Daniel

Grupo 941

Ejercicios 1 Unidad 1

Python

Cesar Antillon Macias

Ensenada B.C. a 24 de febrero de 2024

Neutralidad de genero

Leer el párrafo de texto: Leer el párrafo completo para comprender el contexto y las palabras que necesitan ser reemplazadas.

Preparar la lista de palabras de género y sus reemplazos neutrales: Tener a mano la lista de palabras específicas de género y sus correspondientes términos neutrales. Por ejemplo: "esposa" o "marido" por "cónyuge", "hombre" o "mujer" por "persona", "hija" o "hijo" por "niño", etc.

Identificar las palabras de género en el texto: Leer el párrafo nuevamente y subrayar o resaltar las palabras que necesitan ser reemplazadas según la lista preparada.

Reemplazar manualmente las palabras de género por las neutrales: Una vez identificadas las palabras específicas de género, sustituirlas manualmente por sus términos neutrales correspondientes. Esto se puede hacer utilizando lápiz y papel o un procesador de texto en una computadora.

Revisar el párrafo modificado: Después de realizar los reemplazos, leer nuevamente el párrafo completo para asegurarse de que los cambios se hayan realizado correctamente y que el texto tenga coherencia y sentido.

Realizar ajustes finales si es necesario: Si se nota algún error o inconsistencia en el texto modificado, realizar los ajustes necesarios para corregirlo.

Coches autónomos

a) Usa un código para entrar al taxi sin conductor, y si algo va mal, presiona un botón de emergencia.

b) Los autos autónomos pueden hablar entre ellos para decidir quién pasa primero en un puente de un carril.

c) Si el auto autónomo es multado, la empresa dueña probablemente pague la multa.

d) Si el auto autónomo no se mueve en un semáforo, intenta contactar a la empresa o muévelo si es seguro.

e) Necesitamos seguridad fuerte para evitar que alguien hackee un auto autónomo.

f) Los autos autónomos enfrentarán muchos desafíos nuevos, como lidiar con peatones y encontrar estacionamiento.

Lenguaje ensamblador

Aunque el lenguaje de programación ensamblador es ampliamente considerado obsoleto debido a su complejidad y falta de portabilidad, aún se utiliza en situaciones donde se requiere un control preciso sobre el hardware y se busca maximizar el rendimiento del sistema. Esto incluye el desarrollo de sistemas embebidos, controladores de dispositivos, optimización de rendimiento en aplicaciones críticas, programación de tiempo real y depuración de bajo nivel. En estas circunstancias, el ensamblador ofrece la capacidad de escribir código altamente eficiente y acceder directamente al hardware, lo que lo convierte en la opción preferida para ciertos escenarios de desarrollo de software.

Machine intelligence

Para disipar los temores sobre las máquinas inteligentes en los hogares, es crucial implementar medidas como garantizar transparencia y control de datos, asegurar la seguridad de los dispositivos, proporcionar educación sobre la inteligencia artificial, desarrollar de manera ética la IA, demostrar beneficios tangibles, y fomentar la adaptación y la capacitación de las personas. Estas estrategias pueden ayudar a generar confianza y promover una adopción más amplia de la tecnología inteligente en los hogares.

Machine learning en chip

El aprendizaje automático en chip está siendo aplicado en diversas áreas emergentes, incluyendo IoT y dispositivos inteligentes, vehículos autónomos, robótica y automatización industrial, procesamiento de imágenes, salud y dispositivos médicos, y seguridad y vigilancia. Estos chips permiten procesar datos y tomar decisiones inteligentes de manera local, lo que aumenta la eficiencia y la autonomía de los sistemas en tiempo real.

Proyecto OWASP

OWASP (Open Web Application Security Project) es una comunidad global dedicada a mejorar la seguridad del software. Sus logros incluyen la creación del Top 10 de OWASP, herramientas y recursos de seguridad, conferencias y una comunidad activa. Sin embargo, enfrenta desafíos como la conciencia limitada sobre seguridad, la evolución de amenazas, recursos limitados y la complejidad del panorama de seguridad. A pesar de estos desafíos, OWASP continúa siendo un actor importante en la promoción y mejora de la seguridad del software a nivel mundial.

Desarrollo de aplicaciones móviles con Python

BeeWare: BeeWare es una colección de herramientas para el desarrollo de aplicaciones nativas en Python. Incluye proyectos como Toga (para crear interfaces de usuario multiplataforma), Batavia (para ejecutar código Python en navegadores web), y otros. BeeWare es multiplataforma y permite desarrollar aplicaciones para iOS, Android, Windows, macOS y Linux.

Kivy: Kivy es un framework de código abierto para el desarrollo de aplicaciones multiplataforma en Python. Permite crear aplicaciones con interfaces de usuario flexibles y dinámicas, y es compatible con iOS, Android, Windows, macOS y Linux.

PyMob: PyMob es un framework para el desarrollo de aplicaciones móviles multiplataforma en Python. Utiliza tecnologías web estándar como HTML5, CSS y JavaScript para crear aplicaciones nativas para iOS y Android. PyMob proporciona un conjunto de herramientas para desarrollar, depurar y empaquetar aplicaciones para ambas plataformas.

Pythonista: Pythonista es una aplicación para iOS que permite escribir y ejecutar código Python directamente en dispositivos iOS. Aunque no es una herramienta de desarrollo multiplataforma en sí misma, Pythonista es útil para escribir scripts y prototipos de aplicaciones en Python en dispositivos iOS.

KivyMD: KivyMD es una biblioteca que proporciona widgets y componentes de Material Design para el framework Kivy. Permite desarrollar aplicaciones multiplataforma con interfaces de usuario que siguen las directrices de diseño de Material Design de Google.

Chaquopy: Chaquopy es una herramienta que permite integrar Python en aplicaciones Android escritas en Java o Kotlin. Permite utilizar código Python para agregar funcionalidades específicas a aplicaciones Android existentes.