

1. Escriba un programa en Python que dado una lista de strings tenga como salida los siguientes casos.

- alternate(["abc", "123", "xyz"]) => "a1xb2yc3z"
- alternate(["abc", "1"]) => "a1b c "
- alternate(["abcd", "12", "x"]) => "a1xb2 c d "
- alternate(["ab", "1234"]) => "a1b2 3 4"

2. Escriba un programa en Javascript que dado una lista string tenga como salida los siguientes casos.

- alternate(["abc", "123", "xyz"]) => "a1xb2yc3z"
- alternate(["abc", "1"]) => "a1b c "
- alternate(["abcd", "12", "x"]) => "a1xb2 c d "
- alternate(["ab", "1234"]) => "a1b2 3 4"

3. Escriba una instrucción SQL para encontrar aquellas filas de la tabla testtable que contienen el guión bajo del carácter de escape (_) en su columna 'col1'.

Mesa de muestras: testtable

col1

A001/DJ-402\44_/100/2015
A001_\DJ-402\44_/100/2015
A001_DJ-402-2014-2015
A002_DJ-401-2014-2015
A001/DJ_401
A001/DJ_402\44
A001/DJ_402\44\2015
A001/DJ-402%45\2015/200
A001/DJ_402\45\2015%100
A001/DJ_402%45\2015/300
A001/DJ-402\44

4. Usando el lenguaje Ruby, haga que la función AlphabetSoup (str) tome el parámetro str string que se está pasando y devuelva la cadena con las letras en orden alfabético (es decir, hello se convierte en ehlllo). Suponga que los números y símbolos de puntuación no se incluirán en la cadena. Ejemplos de casos de prueba:

Entrada: "codificador"

Salida: "accddfiioor"

Entrada: "hooplah"

Salida: "ahhloop"

Comprensión en Odoo (Django Devs, pasar a la siguiente sección)

5 - Desarrollar un módulo con las siguientes características, es necesario para una empresa de eventos programar las reservas, y poder observarlas en una vista, a su vez es necesario poder obtener el reporte de las reservas por días con el “Nombre del Cliente”, “Fecha y Hora de Inicio”, “Fecha y Hora Fin”, “Salón Reservado”, “Cantidad de Mesas”, “Cantidad de Sillas” y “Responsable del evento”.

- Una vez finalizado el uso del salón los mismos deben ser marcados como “Disponibles” en el módulo, los estados que podrían tener son los siguientes “En uso”.
- Las reservas tendrán estados y un responsable, las cuales deben contener en su registro tipo de evento, Cliente, responsable(Interno), hora de inicio, hora fin, salón, cantidad de sillas, cantidad de mesas, si tiene o no servicio de catering y el tipo de evento.
- Es necesario poder ver los registros en una vista Kanban al ingresar a ellos en los cuales mostrará el estado y las horas de inicio y fin de cada uno de las reservas, con el nombre del cliente.
- Debe existir un usuario supervisor quien convertirá en facturas cada uno de los servicios una vez confirmada la reserva.
- Es necesario saber cuántas horas durará ocupado cada uno de los salones.

2 - ¿Qué es un campo calculado?

3 - ¿Qué es un campo related?

4 - ¿Nombre al menos 5 decoradores más usados en Odoo?

Compresión en Django (Odoo devs, obviar esta prueba)

6 - Para un cliente es necesario crear un sistema sencillo que permita organizar tareas en un calendario, las mismas deben poseer la capacidad de asignarles prioridades y un checklist, así como especificar si es de tipo Tarea o Reunión, cada tipo debe permitir agregar una observación.

Criterios a evaluar:

- Las tareas no deben solaparse es decir no debe ser posible para el usuario agendar dos tareas en el mismo lapso de tiempo.
- Las tareas deben tener prioridades y tipo (Reunión o Tarea).
- Por defecto todas las tareas se crean en "Pendiente".
- El usuario debe poder marcar a "Listo" las tareas que ya ha realizado.
- Todas deben tener un campo observación.
- Debe ser posible administrar las tareas mediante la RestAPI del Rest Framework de Django.
- Es necesario un reporte que muestre al usuario la efectividad que ha tenido en el día, semana o mes, $\text{Efectividad} = \text{Cant. Tareas Asignadas} / \text{Cant. Tareas Completadas}$.
- Se debe permitir al usuario visualizar las tareas en un calendario o una lista que debe permitir filtrar por días, mes, año, pendientes, completadas, de tipo tarea o de tipo reunión.
- Solamente un usuario autorizado puede realizar operaciones en el website.