## Pruébalo tú mismo

9-6. Puesto de helados: Un puesto de helados es una clase específica de restaurante. Escribe una clase llamada IceCreamStand que herede de la clase Restaurant que escribiste en el Ejercicio 9-1(página 162) o en el Ejercicio 9-4(página 166). Cualquiera de las dos versiones de la clase funcionará; simplemente elige la que más te guste. Añade un atributo llamado flavors que almacene una lista de sabores de helado. Escribe un método que muestre estos sabores. Crea una instancia de IceCreamStand, y llama a este método.

9-7. Administrador: Un administrador es una clase especial de usuario. Escribe una clase llamada Admin que herede de la clase User que escribiste en el Ejercicio 9-3(página 162) o en el Ejercicio 9-5(página 167). Añade un atributo, privileges, que almacene una lista de cadenas como "can add post", "can delete post", "can ban user", etc. Escribe un método llamado show\_privileges() que enumere el conjunto de privilegios del administrador. Crea una instancia de Admin, y llama a tu método.

9-8. Privilegios: Escribe una clase Privileges separada. La clase debe tener un atributo, privileges, que almacene una lista de cadenas como la descrita en el Ejercicio 9-7. Traslada el método show\_privileges() a esta clase. Crea una instancia de Privileges como atributo en la clase Admin. Crea una nueva instancia de Admin y utiliza su método para mostrar sus privilegios.

9-9. Actualización de la Batería: Utiliza la versión final de *electric\_car.py* de esta sección. Añade un método a la clase Battery llamado upgrade\_battery(). Este método debería comprobar el tamaño de la batería y establecer la capacidad en 65 si aún no lo está. Crea un coche eléctrico con un tamaño de batería por defecto, llama a get\_range() una vez y, a continuación, llama a get\_range() una segunda vez después de actualizar la batería. Deberías ver un aumento de la autonomía del coche.

[anterior](c09_18.html)[Subtema 19 de 34: (Ver todo)](c09.html)[siguiente](c09_20.html)