

Método generate_trips()

- Este método se encarga de **crear registros sintéticos de viajes**.
- Cada viaje combina tres entidades clave: un vehículo, un conductor y una ruta.
- El proceso:
 - Selecciona aleatoriamente un vehicle_id, un driver_id y un route_id de las tablas ya generadas.
 - Define una **fecha y hora de salida** (departure_datetime) dentro de un rango temporal (por ejemplo, los últimos dos años).
 - Calcula la **fecha y hora de llegada** (arrival_datetime) sumando una duración aleatoria (entre 1 y 72 horas).
 - Asigna un **peso total transportado** (total_weight_kg) positivo y realista.
- Resultado: un DataFrame con miles de viajes, cada uno consistente y referenciado correctamente a vehículos, conductores y rutas.

Método gethourly_distribution()

- Este método es un **auxiliar estadístico** que define cómo se distribuyen las horas de salida de los viajes.
- En lugar de asignar horas al azar de manera uniforme, genera una **curva de probabilidad por cada hora del día (0–23)**.
- Ejemplo de la distribución:
 - Horas de madrugada (0–5): baja probabilidad.
 - Horas de la mañana (6–11): alta probabilidad, porque es cuando suelen iniciar más viajes.
 - Mediodía y tarde (12–17): probabilidad moderada-alta.
 - Noche (18–23): probabilidad media-baja.
- Esto permite que los datos sintéticos reflejen **patrones más realistas de operación logística**.

Cómo se usan juntos en el script

1. Primero se llama a gethourly_distribution() para obtener el vector de probabilidades por hora.
2. Luego, dentro de generate_trips(), se utiliza ese vector para elegir la hora de salida de cada viaje con random.choices().
3. Así, cada departure_datetime no solo tiene una fecha aleatoria, sino también una hora que sigue la distribución definida.
4. El resultado es un conjunto de viajes que simulan la realidad: más viajes en la mañana y menos en la madrugada, lo que hace que los datos sean más creíbles para análisis posteriores.

Síntesis

- `generate_trips()` → crea los viajes, relacionando vehículos, conductores y rutas, con tiempos y cargas coherentes.
- `gethourly_distribution()` → define la probabilidad de salida por hora del día, para que los viajes reflejen patrones reales.
- **Uso conjunto:** el script genera datos sintéticos que no son completamente aleatorios, sino que imitan la dinámica diaria de una empresa logística.