

```
In [70]: ▶ !pip install simpy
!pip install plotly==5.5.0

Requirement already satisfied: simpy in c:\users\pulpin\anaconda3\lib\site-packages (4.0.1)
Collecting plotly==5.5.0
  Downloading https://files.pythonhosted.org/packages/b3/f4/23d4a698db9fe772f7fdf40ac17b743c4b0d80274732c59db5bd45acb3be/plotly-5.5.0-py2.py3-none-any.whl (https://files.pythonhosted.org/packages/b3/f4/23d4a698db9fe772f7fdf40ac17b743c4b0d80274732c59db5bd45acb3be/plotly-5.5.0-py2.py3-none-any.whl) (26.5MB)
Requirement already satisfied: six in c:\users\pulpin\anaconda3\lib\site-packages (from plotly==5.5.0) (1.12.0)
Collecting tenacity>=6.2.0 (from plotly==5.5.0)
  Downloading https://files.pythonhosted.org/packages/f2/a5/f86bc8d67c979020438c8559cc70cfe3a1643fd160d35e09c9cca6a09189/tenacity-8.0.1-py3-none-any.whl (https://files.pythonhosted.org/packages/f2/a5/f86bc8d67c979020438c8559cc70cfe3a1643fd160d35e09c9cca6a09189/tenacity-8.0.1-py3-none-any.whl)
Installing collected packages: tenacity, plotly
Successfully installed plotly-5.5.0 tenacity-8.0.1

In [100]: ▶ # Imports
import collections
import random
import simpy
import matplotlib.pyplot as plt
import numpy as np
from scipy import stats
import seaborn as sns
from random import randint
from simpy.events import Timeout

In [101]: ▶ NUM_BUSES = 5
NUM_OFICINAS = 1
CAPACIDAD = 40
DIAS = 7
PERSONAS = 65070
TIEMPO_SIMULACION = 720 #720 minutos 12 horas tiempo en segundos
p = 0.5 # parametro de forma
mu = 30 # parametro de forma
```

In [102]:  #CREAMOS NOMBRES ALEATORIOS DE MUJERES

```
data_female=[
    "Lucía",
    "Sofía",
    "Martina",
    "María",
    "Julia",
    "Paula",
    "Valeria",
    "Emma",
    "Daniela",
    "Carla",
    "Alba",
    "Noa",
    "Alma",
    "Sara",
    "Carmen",
    "Vega",
    "Lara",
    "Mia",
    "Valentina",
    "Olivia",
    "Claudia",
    "Jimena",
    "Lola",
    "Chlóe",
    "Aitana",
    "Abril",
    "Ana",
    "Laia",
    "Triana",
    "Candela",
    "Alejandra",
    "Elena",
    "Vera",
    "Manuela",
    "Adriana",
    "Inés",
    "Marta",
    "Carlota",
    "Irene",
    "Victoria",
    "Blanca",
    "Marina",
    "Laura",
    "Rocío",
    "Alicia",
    "Clara",
    "Nora",
    "Lía",
    "Ariadna",
    "Zoe",
    "Amira",
    "Gala",
    "Celia",
    "Leire",
    "Eva",
    "Ángela",
    "Andrea",
    "África",
    "Luna",
    "Ainhua",
    "Ainara",
    "India",
    "Nerea",
    "Ona",
    "Elsa",
    "Isabel",
    "Leyre",
    "Gabriela",
    "Aina",
    "Cayetana",
    "Iria",
    "Jana",
    "Mar",
    "Cloe",
    "Lina",
    "Julieta",
    "Adara",
    "Naia",
    "Iris",
    "Nour",
    "Mara",
    "Helena",
    "Yasmin",
    "Natalia",
    "Arlet",
    "Diana",
    "Aroa",
    "Amaia",
    "Cristina",
    "Nahia",
    "Isabella",
    "Malak",
    "Elia",
    "Carolina",
    "Berta",
    "Fátima",
    "Nuria",
    "Azahara",
    "Macarena",
    "Aurora",
]
#CREAMOS NOMBRES ALEATORIOS DE HOMBRES
data_male=[
    "Hugo",
    "Martín",
    "Lucas",
    "Mateo",
    "Leo",
    "Daniel",
    "Alejandro",
    "Pablo",
    "Manuel",
    "Álvaro",
    "Adrián",
    "David",
    "Mario",
    "Enzo",
    "Diego",
    "Marcos",
    "Izan",
    "Javier",
    "Marco",
    "Álex",
    "Bruno",
```

```
"Oliver",
"Miguel",
"Thiago",
"Antonio",
"Marc",
"Carlos",
"Ángel",
"Juan",
"Gonzalo",
"Gael",
"Sergio",
"Nicolás",
"Dylan",
"Gabriel",
"Jorge",
"José",
"Adam",
"Liam",
"Eric",
"Samuel",
"Darío",
"Héctor",
"Luca",
"Iker",
"Amir",
"Rodrigo",
"Saúl",
"Víctor",
"Francisco",
"Iván",
"Jesús",
"Jaime",
"Aarón",
"Rubén",
"Ian",
"Guillermo",
"Erik",
"Mohamed",
"Julen",
"Luis",
"Pau",
"Unai",
"Rafael",
"Joel",
"Alberto",
"Pedro",
"Raúl",
"Aitor",
"Santiago",
"Rayan",
"Pol",
"Nil",
"Noah",
"Jan",
"Asier",
"Fernando",
"Alonso",
"Matías",
"Biel",
"Andrés",
"Axel",
"Ismael",
"Martí",
"Arnau",
"Imran",
"Luka",
"Ignacio",
"Aleix",
"Alan",
"Elías",
"Omar",
"Isaac",
"Youssef",
"Jon",
"Teo",
"Mauro",
"Óscar",
"Cristian",
"Leonardo"
]
```

In []:

In [116]:

```
data_bus = [
{
  "code": "01",
  "name": "Turismo Oriental",
  "origen": "Cuenca",
  "destination": "Quito"
},
{
  "code": "02",
  "name": "Santa",
  "origen": "Cuenca",
  "destination": "Guayaquil"
},
{
  "code": "03",
  "name": "Super Semeria",
  "origen": "Quito",
  "destination": "Ambato"
},
{
  "code": "04",
  "name": "Trans Esmeraldas",
  "origen": "Quito",
  "destination": "Cuenca"
},
{
  "code": "05",
  "name": "viajero Internacional",
  "origen": "Cuenca",
  "destination": "Loja"
}
]
```

```
In [117]: ▶ def random_age():  
  
    # Graficando Poisson  
    poisson = stats.poisson(mu) # Distribución  
    # histograma  
    aleatorios = poisson.rvs(1) # genera aleatorios  
  
    return aleatorios[0]
```

```
In [118]: ▶ def customer_object(name, genere, age):  
    return {  
        "name":name,  
        "genere":genere,  
        "age": age,  
    }
```

```
In [119]: ▶ def random_customer():  
    bernoulli = stats.bernoulli(p)  
    aleatorios = bernoulli.rvs(1)  
    idx= randint(0, 99)  
    age = random_age()  
    if aleatorios[0] == 1:  
        return customer_object(data_male[idx], "Male", age)  
    if aleatorios[0] == 0:  
        return customer_object(data_female[idx], "Female", age)
```

```
In [120]: ▶ def generate_time_normal():  
    mu, sigma = 10, 2 # media y desvio estandar  
    normal = stats.norm(mu, sigma)  
    aleatorios = normal.rvs(1) # genera aleatorios  
    return aleatorios[0]
```

```
In [131]: class ABC_Agencia():
def __init__(self, env):
    self.env = env
    self.transporte = simpy.Resource(env, capacity=NUM_BUSES) # Generamos el dispensador 2, Las estaciones de carga
    self.oficinas = simpy.Resource(env, capacity=NUM_OFICINAS) # oficina para venta de tikets
    self.destinos = data_destination
    self.buses = data_bus
    self.vieajes = []
    self.buses_salida = []
    self.boleteria = {
        "boletos":[],
        "num_renegados": 0
    }
    self.load_data()
    self.dispatch = env.process(self.dispatch_buss())

def load_data(self):
    for bus in self.buses:
        bus["container"] = simpy.Container(env, init=CAPACIDAD, capacity=CAPACIDAD)
        bus["sold_out"] = env.event()
        bus["tiempo_agotado"] = None

def dispatch_buss(self):
    while True:
        with self.transporte.request() as transporte:
            yield transporte
            for boleto in self.boleteria['boletos']:
                destino_list = list(filter(lambda destino: destino['destination'] == boleto["bus"]["destination"], self.destinos))[0]
                bus = list(filter(lambda bus: bus['code'] == boleto["bus"]["code"], self.buses))[0]
                if bus in self.buses_salida:
                    continue
                self.buses_salida.append(bus)
                #print("salida", len(self.buses_salida))
                self.vieajes.append(bus)
                yield env.timeout(3600)
                self.env.process(self.return_bus())
                code = boleto["bus"]['code']
                destino = boleto["bus"]['destination']
                pasajeros = CAPACIDAD - bus["container"].level
                print(f'Sale el bus N° {code} hasta {destino} con {pasajeros} pasajeros')

def return_bus(self):
    while True:
        with self.transporte.request() as transporte:
            yield transporte
            for boleto in self.boleteria['boletos']:
                destino_list = list(filter(lambda destino: destino['destination'] == boleto["bus"]["destination"], self.destinos))[0]
                bus = list(filter(lambda bus: bus['code'] == boleto["bus"]["code"], self.buses))[0]
                yield env.timeout(destino_list["time"]*2)
                yield bus["container"].put(CAPACIDAD)
                code = boleto["bus"]['code']
                pasajeros = CAPACIDAD - bus["container"].level
                print(f'Regres a el bus N° {code} a la terminal de cuenca con {pasajeros} pasajeros')
                self.buses_salida = [i for i in self.buses_salida if not (i['code'] == bus['code'])]
                #print("Llegada", len(self.buses_salida))
                self.transporte.release(transporte)

def buy_ticket(self, customer):
    with self.oficinas.request() as oficina:
        resultado = yield oficina
        #print("resultado", resultado)
        #if oficina not in resultado:
            #self.boleteria["num_renegados"] += 1
            #return
        tiempo_estimado_compra = generate_time_normal()

    buses = list(filter(lambda bus: bus['destination'] == customer["destination"], self.buses))
    #print(buses)
    for bus in buses:
        if bus["container"].level < customer["num_boletos"]:
            #yield env.timeout(0.5)
            continue
        bus["container"].get(customer["num_boletos"])
        boleto = {
            "customer": customer,
            "destination":list(filter(lambda destino: destino['destination'] == customer["destination"], self.destinos))[0],
            "bus":bus,
            "tiempo_compra": tiempo_estimado_compra
        }
        self.boleteria["boletos"].append(boleto)
        print('N° de boletos: %s, %s, destino: %s, bus N° %s, cooperativa: %s, asientos restantes: %s' %(str(customer["num_boletos"]), boleto["customer"]["name"], boleto["desti
        break
    yield env.timeout(tiempo_estimado_compra)
    self.oficinas.release(oficina)

def arrive_client(self):
    for i in range(PERSONAS):
        destino = random.choice(self.destinos)
        customer = random_customer()
        customer["num_boletos"] = random.randint(1, 6)
        customer["destination"] = destino["destination"]

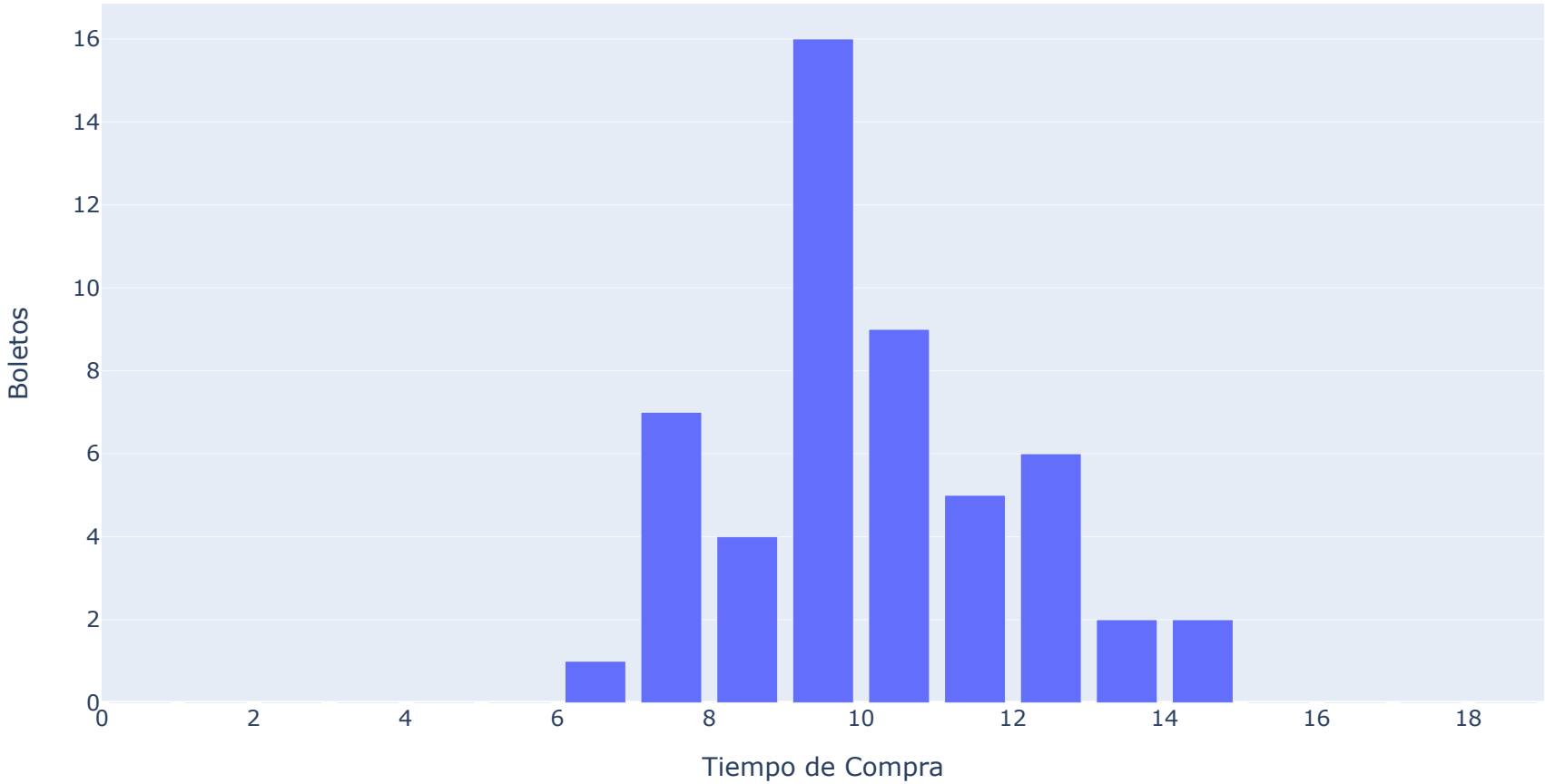
    env.process(self.buy_ticket(customer))
    yield env.timeout(1)
```

```
In [132]: env = simpy.Environment()
          agaencia = ABC_Agencia(env)
          generado = env.process(agaencia.arrive_client())
          env.run(until=TIEMPO_SIMULACION)
```

Nº de boletos: 4, Lía, destino: Quito, bus Nº 01, cooperativa: Turismo Oriental, asientos restantes: 36
Nº de boletos: 3, Julen, destino: Loja, bus Nº 05, cooperativa: viajero Internacional, asientos restantes: 37
Nº de boletos: 5, Cloe, destino: Cuenca, bus Nº 04, cooperativa: Trans Esmeraldas, asientos restantes: 35
Nº de boletos: 4, Ángela, destino: Guayaquil, bus Nº 02, cooperativa: Santa, asientos restantes: 36
Nº de boletos: 1, Claudia, destino: Loja, bus Nº 05, cooperativa: viajero Internacional, asientos restantes: 36
Nº de boletos: 2, Azahara, destino: Cuenca, bus Nº 04, cooperativa: Trans Esmeraldas, asientos restantes: 33
Nº de boletos: 4, Gonzalo, destino: Loja, bus Nº 05, cooperativa: viajero Internacional, asientos restantes: 32
Nº de boletos: 6, Martina, destino: Cuenca, bus Nº 04, cooperativa: Trans Esmeraldas, asientos restantes: 27
Nº de boletos: 4, Carmen, destino: Loja, bus Nº 05, cooperativa: viajero Internacional, asientos restantes: 28
Nº de boletos: 4, Valentina, destino: Guayaquil, bus Nº 02, cooperativa: Santa, asientos restantes: 32
Nº de boletos: 6, Eva, destino: Cuenca, bus Nº 04, cooperativa: Trans Esmeraldas, asientos restantes: 21
Nº de boletos: 4, Jon, destino: Quito, bus Nº 01, cooperativa: Turismo Oriental, asientos restantes: 32
Nº de boletos: 6, Marc, destino: Quito, bus Nº 01, cooperativa: Turismo Oriental, asientos restantes: 26
Nº de boletos: 5, Nil, destino: Loja, bus Nº 05, cooperativa: viajero Internacional, asientos restantes: 23
Nº de boletos: 6, Elia, destino: Cuenca, bus Nº 04, cooperativa: Trans Esmeraldas, asientos restantes: 15
Nº de boletos: 1, Matías, destino: Ambato, bus Nº 03, cooperativa: Super Semeria, asientos restantes: 39
Nº de boletos: 6, Jimena, destino: Guayaquil, bus Nº 02, cooperativa: Santa, asientos restantes: 26
Nº de boletos: 2, Noa, destino: Ambato, bus Nº 03, cooperativa: Super Semeria, asientos restantes: 37
Nº de boletos: 2, Cloe, destino: Loja, bus Nº 05, cooperativa: viajero Internacional, asientos restantes: 21
Nº de boletos: 1, Adriana, destino: Loja, bus Nº 05, cooperativa: viajero Internacional, asientos restantes: 20
Nº de boletos: 1, Lina, destino: Guayaquil, bus Nº 02, cooperativa: Santa, asientos restantes: 25
Nº de boletos: 5, Mario, destino: Quito, bus Nº 01, cooperativa: Turismo Oriental, asientos restantes: 21
Nº de boletos: 3, Adrián, destino: Quito, bus Nº 01, cooperativa: Turismo Oriental, asientos restantes: 18
Nº de boletos: 1, Valentina, destino: Guayaquil, bus Nº 02, cooperativa: Santa, asientos restantes: 24
Nº de boletos: 1, Malak, destino: Cuenca, bus Nº 04, cooperativa: Trans Esmeraldas, asientos restantes: 14
Nº de boletos: 6, Isabel, destino: Guayaquil, bus Nº 02, cooperativa: Santa, asientos restantes: 18
Nº de boletos: 4, Samuel, destino: Ambato, bus Nº 03, cooperativa: Super Semeria, asientos restantes: 33
Nº de boletos: 4, Cristina, destino: Cuenca, bus Nº 04, cooperativa: Trans Esmeraldas, asientos restantes: 10
Nº de boletos: 4, Ona, destino: Loja, bus Nº 05, cooperativa: viajero Internacional, asientos restantes: 16
Nº de boletos: 1, Aurora, destino: Loja, bus Nº 05, cooperativa: viajero Internacional, asientos restantes: 15
Nº de boletos: 1, Ainhoa, destino: Ambato, bus Nº 03, cooperativa: Super Semeria, asientos restantes: 32
Nº de boletos: 5, Julia, destino: Ambato, bus Nº 03, cooperativa: Super Semeria, asientos restantes: 27
Nº de boletos: 5, Chlóe, destino: Guayaquil, bus Nº 02, cooperativa: Santa, asientos restantes: 13
Nº de boletos: 1, Iria, destino: Ambato, bus Nº 03, cooperativa: Super Semeria, asientos restantes: 26
Nº de boletos: 5, Marta, destino: Quito, bus Nº 01, cooperativa: Turismo Oriental, asientos restantes: 13
Nº de boletos: 6, Mara, destino: Loja, bus Nº 05, cooperativa: viajero Internacional, asientos restantes: 9
Nº de boletos: 2, Adam, destino: Guayaquil, bus Nº 02, cooperativa: Santa, asientos restantes: 11
Nº de boletos: 4, Aurora, destino: Loja, bus Nº 05, cooperativa: viajero Internacional, asientos restantes: 5
Nº de boletos: 3, Inés, destino: Guayaquil, bus Nº 02, cooperativa: Santa, asientos restantes: 8
Nº de boletos: 2, Eric, destino: Quito, bus Nº 01, cooperativa: Turismo Oriental, asientos restantes: 11
Nº de boletos: 6, Eric, destino: Quito, bus Nº 01, cooperativa: Turismo Oriental, asientos restantes: 5
Nº de boletos: 1, Julia, destino: Guayaquil, bus Nº 02, cooperativa: Santa, asientos restantes: 7
Nº de boletos: 2, Martí, destino: Loja, bus Nº 05, cooperativa: viajero Internacional, asientos restantes: 3
Nº de boletos: 3, Eric, destino: Loja, bus Nº 05, cooperativa: viajero Internacional, asientos restantes: 0
Nº de boletos: 6, Alma, destino: Guayaquil, bus Nº 02, cooperativa: Santa, asientos restantes: 1
Nº de boletos: 5, Cayetana, destino: Cuenca, bus Nº 04, cooperativa: Trans Esmeraldas, asientos restantes: 5
Nº de boletos: 2, Gonzalo, destino: Cuenca, bus Nº 04, cooperativa: Trans Esmeraldas, asientos restantes: 3
Nº de boletos: 5, Adriana, destino: Ambato, bus Nº 03, cooperativa: Super Semeria, asientos restantes: 21
Nº de boletos: 2, Jana, destino: Cuenca, bus Nº 04, cooperativa: Trans Esmeraldas, asientos restantes: 1
Nº de boletos: 2, David, destino: Ambato, bus Nº 03, cooperativa: Super Semeria, asientos restantes: 19
Nº de boletos: 3, Omar, destino: Ambato, bus Nº 03, cooperativa: Super Semeria, asientos restantes: 16
Nº de boletos: 5, Óscar, destino: Quito, bus Nº 01, cooperativa: Turismo Oriental, asientos restantes: 0

```
In [133]: import plotly.express as px
          #print([boleto["tiempo_compra"] for boleto in agaencia.boleteria["boletos"]])
          counts, bins = np.histogram([boleto["tiempo_compra"] for boleto in agaencia.boleteria["boletos"]], bins=range(0, 20, 1))
          bins = 0.5 * (bins[:-1] + bins[1:])

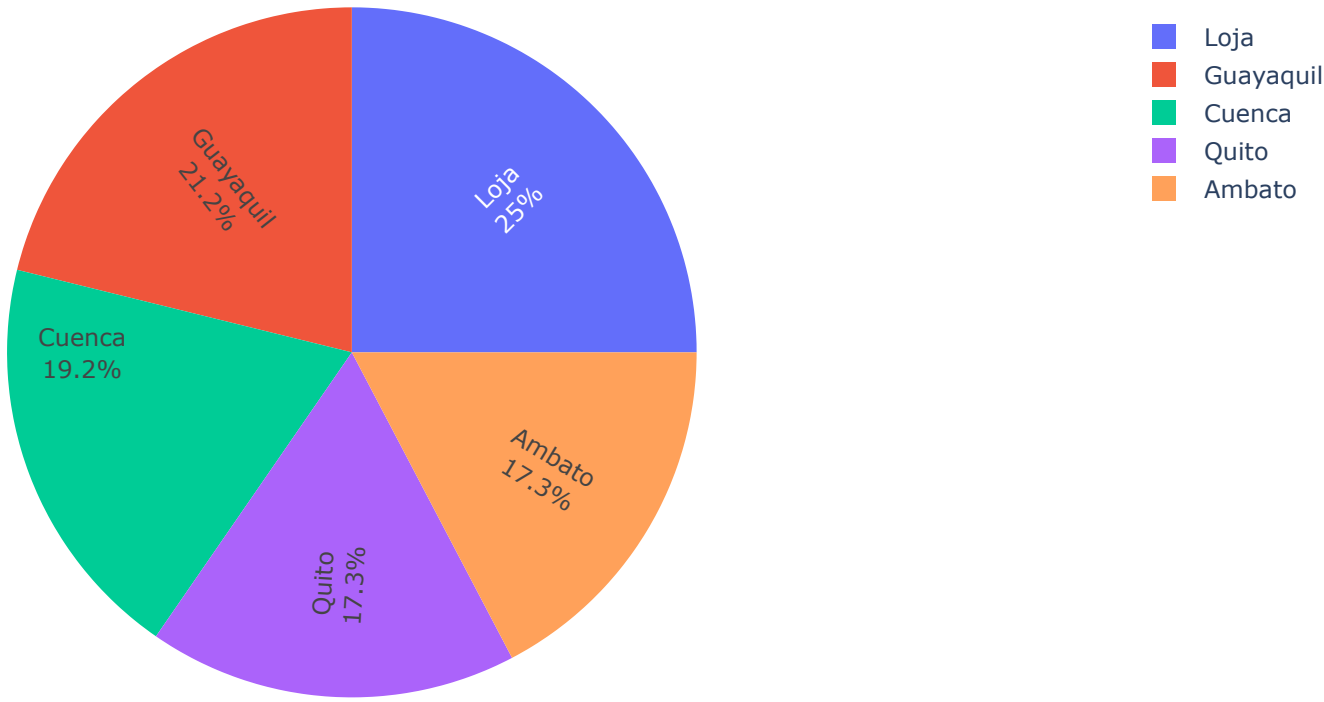
          fig = px.bar(x=bins, y=counts, labels={'x': 'Tiempo de Compra', 'y': 'Boletos'})
          fig.show()
```



```
In [134]: # Lugares mas visitados
import plotly.graph_objects as go
boleto_destino = [boleto["customer"]["destination"] for boleto in agaencia.boleteria["boletos"]]
dest = [destinantion["destination"] for destinantion in data_destination]
count = [sum(x == destinantion for x in boleto_destino) for destinantion in dest]

#print(boleto_destino)
#print(dest)
#print(count)
fig = go.Figure(data=[go.Pie(labels=dest, values=count, textinfo='label+percent',
                             insidetextorientation='radial'
                             )])

fig.show()
```



```
In [ ]:
```

```
In [ ]:
```