Safertek backend assignment: 2100031993

1)Write a query to find the address (location\_id, street\_address, city, state\_province, country\_name) of Canada using join in Python (Input -We Can create a table in FrontEnd like using variables)

Code:

import pandas as pd  
  
# Data for locations  
locations = pd.DataFrame([  
 {"location\_id": 1000, "street\_address": "1297 Via Cola di Rie", "postal\_code": "989", "city": "Roma", "state\_province": "", "country\_id": "IT"},  
 {"location\_id": 1100, "street\_address": "93091 Calle della Te", "postal\_code": "10934", "city": "Venice", "state\_province": "", "country\_id": "IT"},  
 {"location\_id": 1200, "street\_address": "2017", "postal\_code": "1689", "city": "Shinjuku-ku", "state\_province": "Tokyo Prefecture", "country\_id": "JP"},  
 {"location\_id": 1300, "street\_address": "9450 Kamiya-cho", "postal\_code": "6823", "city": "Hiroshima", "state\_province": "", "country\_id": "JP"},  
 {"location\_id": 1400, "street\_address": "2014 Jabberwocky Rd", "postal\_code": "26192", "city": "Southlake", "state\_province": "Texas", "country\_id": "US"},  
 {"location\_id": 1500, "street\_address": "2011 Interiors Blvd", "postal\_code": "99236", "city": "South San Francisco", "state\_province": "California", "country\_id": "US"},  
 {"location\_id": 1600, "street\_address": "2007 2004 Charade Rd", "postal\_code": "50090", "city": "South Brunswick", "state\_province": "New Jersey", "country\_id": "US"},  
 {"location\_id": 1700, "street\_address": "98199 Seattle Blvd", "postal\_code": "98199", "city": "Seattle", "state\_province": "Washington", "country\_id": "US"},  
 {"location\_id": 1800, "street\_address": "147 Spadina Ave", "postal\_code": "MSV 2L7", "city": "Toronto", "state\_province": "Ontario", "country\_id": "CA"}  
])  
  
# Data for countries  
countries = pd.DataFrame([  
 {"country\_id": "AR", "country\_name": "Argentina", "region\_id": 2},  
 {"country\_id": "AU", "country\_name": "Australia", "region\_id": 3},  
 {"country\_id": "BE", "country\_name": "Belgium", "region\_id": 1},  
 {"country\_id": "BR", "country\_name": "Brazil", "region\_id": 2},  
 {"country\_id": "CA", "country\_name": "Canada", "region\_id": 2},  
 {"country\_id": "CH", "country\_name": "Switzerland", "region\_id": 1},  
 {"country\_id": "CN", "country\_name": "China", "region\_id": 3},  
 {"country\_id": "DE", "country\_name": "Germany", "region\_id": 1}  
])  
  
# Join on country\_id to get only Canadian addresses  
result = pd.merge(locations, countries[countries['country\_name'] == "Canada"], on="country\_id", how="inner")  
result = result[["location\_id", "street\_address", "city", "state\_province", "country\_name"]]  
print(result)

2) Write a query to find the address (location\_id, street\_address, city, state\_province, coutry\_name) of Canada NOT using join in Python. (Input -We Can create a table in FrontEnd like using variables)  
code: # Data for locations  
locations = [  
 {"location\_id": 1000, "street\_address": "1297 Via Cola di Rie", "postal\_code": "989", "city": "Roma", "state\_province": "", "country\_id": "IT"},  
 {"location\_id": 1100, "street\_address": "93091 Calle della Te", "postal\_code": "10934", "city": "Venice", "state\_province": "", "country\_id": "IT"},  
 {"location\_id": 1200, "street\_address": "2017", "postal\_code": "1689", "city": "Shinjuku-ku", "state\_province": "Tokyo Prefecture", "country\_id": "JP"},  
 {"location\_id": 1300, "street\_address": "9450 Kamiya-cho", "postal\_code": "6823", "city": "Hiroshima", "state\_province": "", "country\_id": "JP"},  
 {"location\_id": 1400, "street\_address": "2014 Jabberwocky Rd", "postal\_code": "26192", "city": "Southlake", "state\_province": "Texas", "country\_id": "US"},  
 {"location\_id": 1500, "street\_address": "2011 Interiors Blvd", "postal\_code": "99236", "city": "South San Francisco", "state\_province": "California", "country\_id": "US"},  
 {"location\_id": 1600, "street\_address": "2007 2004 Charade Rd", "postal\_code": "50090", "city": "South Brunswick", "state\_province": "New Jersey", "country\_id": "US"},  
 {"location\_id": 1700, "street\_address": "98199 Seattle Blvd", "postal\_code": "98199", "city": "Seattle", "state\_province": "Washington", "country\_id": "US"},  
 {"location\_id": 1800, "street\_address": "147 Spadina Ave", "postal\_code": "MSV 2L7", "city": "Toronto", "state\_province": "Ontario", "country\_id": "CA"}  
]  
  
# Data for countries  
countries = [  
 {"country\_id": "AR", "country\_name": "Argentina", "region\_id": 2},  
 {"country\_id": "AU", "country\_name": "Australia", "region\_id": 3},  
 {"country\_id": "BE", "country\_name": "Belgium", "region\_id": 1},  
 {"country\_id": "BR", "country\_name": "Brazil", "region\_id": 2},  
 {"country\_id": "CA", "country\_name": "Canada", "region\_id": 2},  
 {"country\_id": "CH", "country\_name": "Switzerland", "region\_id": 1},  
 {"country\_id": "CN", "country\_name": "China", "region\_id": 3},  
 {"country\_id": "DE", "country\_name": "Germany", "region\_id": 1}  
]  
  
# Find Canadian country ID  
canadian\_country\_id = [country['country\_id'] for country in countries if country['country\_name'] == "Canada"][0]  
  
# Find locations in Canada  
canadian\_locations = [location for location in locations if location['country\_id'] == canadian\_country\_id]  
  
# Print results  
for location in canadian\_locations:  
 print(f"Location ID: {location['location\_id']}, Address: {location['street\_address']}, City: {location['city']}, State: {location['state\_province']}, Country: Canada")