Install Required Libraries

```
!pip install requests pandas google-cloud-bigquery
```

```
Requirement already satisfied: requests in /usr/local/lib/python3.11/dist-packages (2.32.3)
Requirement already satisfied: pandas in /usr/local/lib/python3.11/dist-packages (2.2.2)
Requirement already satisfied: google-cloud-bigquery in /usr/local/lib/python3.11/dist-packages (3.34.0)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.11/dist-packages (from requests) (3.4.
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.11/dist-packages (from requests) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.11/dist-packages (from requests) (2.4.0)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.11/dist-packages (from requests) (2025.6.15)
Requirement already satisfied: numpy>=1.23.2 in /usr/local/lib/python3.11/dist-packages (from pandas) (2.0.2)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.11/dist-packages (from pandas) (2.9.0.po
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.11/dist-packages (from pandas) (2025.2)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.11/dist-packages (from pandas) (2025.2)
Requirement already satisfied: google-api-core<3.0.0,>=2.11.1 in /usr/local/lib/python3.11/dist-packages (from google-ap
Requirement already satisfied: google-auth<3.0.0,>=2.14.1 in /usr/local/lib/python3.11/dist-packages (from google-cloud-
Requirement already satisfied: google-cloud-core<3.0.0,>=2.4.1 in /usr/local/lib/python3.11/dist-packages (from google-c
Requirement already satisfied: google-resumable-media<3.0.0,>=2.0.0 in /usr/local/lib/python3.11/dist-packages (from goo
Requirement already satisfied: packaging>=24.2.0 in /usr/local/lib/python3.11/dist-packages (from google-cloud-bigquery)
Requirement already satisfied: googleapis-common-protos<2.0.0,>=1.56.2 in /usr/local/lib/python3.11/dist-packages (from
Requirement already satisfied: protobuf!=3.20.0,!=3.20.1,!=4.21.0,!=4.21.1,!=4.21.2,!=4.21.3,!=4.21.4,!=4.21.5,<7.0.0,>=
Requirement already satisfied: proto-plus<2.0.0,>=1.22.3 in /usr/local/lib/python3.11/dist-packages (from google-api-cor
Requirement already satisfied: grpcio<2.0.0,>=1.33.2 in /usr/local/lib/python3.11/dist-packages (from google-api-core[gr
Requirement already satisfied: grpcio-status<2.0.0,>=1.33.2 in /usr/local/lib/python3.11/dist-packages (from google-api-
Requirement already satisfied: cachetools<6.0,>=2.0.0 in /usr/local/lib/python3.11/dist-packages (from google-auth<3.0.0
Requirement already satisfied: pyasn1-modules>=0.2.1 in /usr/local/lib/python3.11/dist-packages (from google-auth<3.0.0,
Requirement already satisfied: rsa<5,>=3.1.4 in /usr/local/lib/python3.11/dist-packages (from google-auth<3.0.0,>=2.14.1
Requirement already satisfied: google-crc32c<2.0dev,>=1.0 in /usr/local/lib/python3.11/dist-packages (from google-resuma
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.11/dist-packages (from python-dateutil>=2.8.2->pandas)
Requirement already satisfied: pyasn1<0.7.0,>=0.6.1 in /usr/local/lib/python3.11/dist-packages (from pyasn1-modules>=0.2
```

```
import requests
import pandas as pd
from datetime import datetime
```

```
#Creating a Simple Data Collector Agent...

def data_collector_agent(country="India"):
    url = f"https://disease.sh/v3/covid-19/countries/{country}"
    response = requests.get(url)

    if response.status_code == 200:
        data = response.json()
        df = pd.DataFrame([{
            "country": data["country"],
            "cases": data["cases"],
            "todayCases": data["todayCases"],
            "deaths": data["deaths"],
            "todayDeaths": data["todayDeaths"],
            "recovered": data["recovered"],
            "active": data["active"],
            "updated": datetime.fromtimestamp(data["updated"]/1000)
        }])
        print("Data collected successfully!")
        return df
    else:
        print("Failed to fetch data.")
        return None
```

```
# Run the Agent..

df = data_collector_agent("India")
df
```

```
Data collected successfully!
   country     cases  todayCases  deaths  todayDeaths  recovered    active                  updated
0    India  45035393           0  533570            0          0  44501823  2025-06-23 18:14:36.403
```

```
# Data cleaning..
def data_cleaner_agent(df):
    # 1. Drop unnecessary columns
    if 'recovered' in df.columns:
        df = df.drop(columns=['recovered'])
    # 2. Handle missing values (if any)
    df = df.fillna(0)

    # 3. Ensure correct data types
    df['cases'] = df['cases'].astype(int)
    df['todayCases'] = df['todayCases'].astype(int)
    df['deaths'] = df['deaths'].astype(int)
    df['todayDeaths'] = df['todayDeaths'].astype(int)
    df['active'] = df['active'].astype(int)
    df['updated'] = pd.to_datetime(df['updated'])

    print("Data cleaned successfully!")
    return df


# Running the Cleaner Agent..

clean_df = data_cleaner_agent(df)
clean_df
```

```
Data cleaned successfully!
```

	country	cases	todayCases	deaths	todayDeaths	active	updated
0	India	45035393	0	533570	0	44501823	2025-06-23 18:14:36.403

```
# TrendAnalyzerAgent.......

def trend_analyzer_agent(df):
    trend_summary = []

    for index, row in df.iterrows():
        trend = "Stable"

        if row["todayCases"] > 10000:
            trend = "🚨 Major Spike Detected"
        elif row["todayCases"] > 1000:
            trend = "⚠️ Moderate Spike Detected"
        elif row["todayCases"] > 0:
            trend = "🔍 Slight Increase Observed"
        elif row["todayCases"] == 0:
            trend = "✅ No new cases today"

        summary = {
            "country": row["country"],
            "date": row["updated"].date(),
            "total_cases": row["cases"],
            "today_cases": row["todayCases"],
            "status": trend
        }
        trend_summary.append(summary)

    result_df = pd.DataFrame(trend_summary)
    print("Trend analysis complete.")
    return result_df



# Run the Agent:....

trend_df = trend_analyzer_agent(clean_df)
trend_df
```

```
Trend analysis complete.
```

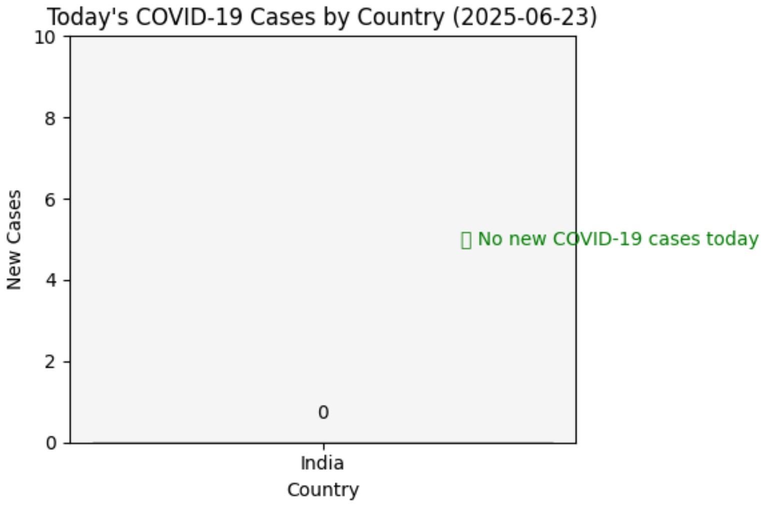
	country	date	total_cases	today_cases	status
0	India	2025-06-23	45035393	0	✅ No new cases today

```
# Next Agent: InsightGeneratorAgent.....
def insight_generator_agent(trend_df):
    insights = []

    for index, row in trend_df.iterrows():
        message = (
            f"As of {row['date']}, {row['country']} reported "
            f"{row['today_cases']} new COVID-19 case(s). "
            f"The total number of confirmed cases is {row['total_cases']}. "
            f"Status: {row['status']}."
        )
        insights.append(message)

    print("Insight generation complete.")
    return insights



# Running the agent....
insight_texts = insight_generator_agent(trend_df)

# Display insights
for insight in insight_texts:
    print("📝", insight)
```

```
Insight generation complete.
📝 As of 2025-06-23, India reported 0 new COVID-19 case(s). The total number of confirmed cases is 45035393. Status: ✅
```

```
import matplotlib.pyplot as plt

def dashboard_agent_colab(trend_df):
    today_cases = trend_df['today_cases'].iloc[0]
    country = trend_df['country'].iloc[0]
    date = trend_df['date'].iloc[0]

    fig, ax = plt.subplots(figsize=(6, 4))
    ax.set_facecolor('#f9f9f9')  # light background

    bars = ax.bar([country], [today_cases], color='lightgreen', edgecolor='black')

    ax.set_title(f"Today's COVID-19 Cases by Country ({date})")
    ax.set_ylabel("New Cases")
    ax.set_xlabel("Country")

    # Adjust y-limit for better visual feedback
    upper_limit = 10 if today_cases == 0 else today_cases + 10
    plt.ylim(0, upper_limit)

    # Show bar label slightly above base even if value is 0
    for bar in bars:
        yval = bar.get_height()
        position = 0.5 if yval == 0 else yval + 0.5
        ax.text(bar.get_x() + bar.get_width() / 2, position, str(yval), ha='center', va='bottom')

    # Optional: Add text for "No data" if everything is zero
    if today_cases == 0:
        ax.text(0.5, 5, "✅ No new COVID-19 cases today", fontsize=10, ha='center', va='center', color='green')

    plt.tight_layout()
    plt.show()


dashboard_agent_colab(trend_df)
```

```
/tmp/ipython-input-15-1444442239.py:31: UserWarning: Glyph 9989 (\N{WHITE HEAVY CHECK MARK}) missing from font(s) DejaVu
  plt.tight_layout()
/usr/local/lib/python3.11/dist-packages/IPython/core/pylabtools.py:151: UserWarning: Glyph 9989 (\N{WHITE HEAVY CHECK MA
  fig.canvas.print_figure(bytes_io, **kw)
```


Today's COVID-19 Cases by Country (2025-06-23)
New Cases
□ No new COVID-19 cases today
India
Country