MLOps With Google Cloud Platform - GCP

What is GCP:
Google Cloud Platform (GCP) is a suite of cloud computing services provided by Google. It allows you to use Google's infrastructure to build, deploy, and scale applications, websites, and services. GCP offers a range of services, including computing power, storage options, data analytics, machine learning tools, and more, all accessible over the internet. This means you can run your projects on Google's powerful hardware without needing to maintain your own servers.

Create GCP Project:

Setting GCP CLI and Set Project:
Commands:

- *gcloud auth login*
- *gcloud projects list*
- *gcloud config set project graphic-adapter-425011-f4*
- *gcloud projects create PROJECT_ID --name="PROJECT_NAME"*
- *gcloud projects delete PROJECT_ID*

GCP Bucket:
Commands:

- **List Buckets:** *gsutil ls*
- **Create Bucket:** *gsutil mb gs://your-bucket-name*
- **Delete Bucket:** *gsutil rb gs://your-bucket-name*
- **Upload file to Bucket:** *gsutil cp local-file.txt gs://your-bucket-name*
- **Download file from Bucket:** *gsutil cp gs://your-bucket-name/remote-file.txt local-file.txt*
- **Copy files within Cloud Storage:** *gsutil cp gs://your-bucket-name/source-file.txt gs://your-bucket-name/destination-file.txt*
- **List Files: List files in a bucket or directory:** *gsutil ls gs://your-bucket-name*
- **Delete Files: Delete a file from a bucket:** *gsutil rm gs://your-bucket-name/remote-file.txt*
- **Set Bucket Permissions: Apply a predefined ACL to a bucket:** *gsutil acl set private gs://your-bucket-name*
- **Set File Permissions: Apply a predefined ACL to a file:** *gsutil acl set public-read gs://your-bucket-name/remote-file.txt*
- **View Bucket or File Permissions: Display ACL for a bucket or file:**
    - *gsutil acl get gs://your-bucket-name*
    - *gsutil acl get gs://your-bucket-name/remote-file.txt*
- **Sync Local Directory to Bucket: Synchronize a local directory with a bucket:**
    - *gsutil rsync -r local-directory gs://your-bucket-name*
- **Set Lifecycle Management Policy: Apply a lifecycle management policy to a bucket:**
    - *gsutil lifecycle set policy.json gs://your-bucket-name*
- **Enable Object Versioning: Enable versioning for a bucket:**
    - *gsutil versioning set on gs://your-bucket-name*
- **Disable Object Versioning: Disable versioning for a bucket:**
    - *gsutil versioning set off gs://your-bucket-name*
- **Check Bucket Usage: Get information about bucket usage:**

    - *gsutil du -s gs://your-bucket-name*
- **View Object Metadata: Display metadata for an object:**
    - *gsutil stat gs://your-bucket-name/remote-file.txt*

Big Query:

Commands

- **Listing Datasets:**
    - *bq ls*
- **List all the dbs in specific project:**
    - *bq ls --project_id=PROJECT_ID*
- **Creating a Dataset:**
    - *bq mk my_dataset*
- **Deleting a Dataset:**
    - *bq rm -r -d my_dataset (The -r flag ensures that all tables in the dataset are removed.*
- **Listing Tables in a Dataset:**
    - *bq ls my_dataset*
- **Creating a Table:**
    - *bq mk --table my_dataset.my_table column1:STRING,column2:INTEGER*
- **Deleting a Table:**
    - *bq rm -t my_dataset.my_table*
- **Inserting Data into a Table:**
    - *bq load --source_format=CSV my_dataset.my_table path_to_csv_file*
- **Querying Data:**
    - *bq query "SELECT column1, column2 FROM my_dataset.my_table WHERE column2 > 10"*
- **Exporting Query Results:**
    - *bq query --format=csv --use_legacy_sql=false "SELECT column1, column2 FROM my_dataset.my_table WHERE column2 > 10" > results.csv*
- **Creating a Partitioned Table:**
    - *bq mk --table --time_partitioning_field column1 my_dataset.my_partitioned_table column1:TIMESTAMP,column2:STRING*
- **Loading Data from Google Cloud Storage:**
    - *bq load --source_format=CSV my_dataset.my_table gs://my_bucket/path_to_csv_file*
- **Queries:**
    - **Show DBS:** *SELECT schema_name FROM INFORMATION_SCHEMA.SCHEMATA;*
    - **Show tables:** *SELECT table_name FROM `my_dataset.INFORMATION_SCHEMA.TABLES`;*
    - **Show table schema:** *SELECT column_name, data_type FROM `my_dataset.INFORMATION_SCHEMA.COLUMNS` WHERE table_name = 'my_table';*
    - **DB statistics:** *SELECT table_name, row_count, size_bytes FROM `my_dataset.INFORMATION_SCHEMA.TABLES`;*
    - **Create non-partitioned table from CSV file:** *CREATE OR REPLACE EXTERNAL TABLE my_dataset.my_external_table PARTITION BY DATE(transaction_date)*

OPTIONS (
format = 'CSV',
uris = ['gs://my_bucket/path/to/file.csv'],
skip_leading_rows = 1
);

- **Creating table from parquet file:**
  CREATE OR REPLACE EXTERNAL TABLE
  my_dataset.my_external_parquet_table
  OPTIONS (
  format = 'PARQUET',
  uris = ['gs://my_bucket/path/to/file.parquet']
  );
- **Update does not support but can use create or replace**

- **Query Optimization?**
- **Scheduling Queries**
- **Big Query Machine Learning**
  - **Classification**
    - **Create Model**

```
CREATE OR REPLACE MODEL `my_dataset.purchase_classification_model`
   OPTIONS(
     model_type='logistic_reg',
     input_label_cols=['purchase']
   ) AS
   SELECT
     age,
     income,
     browsing_time,
     purchase
   FROM
     `my_dataset.customer_data`;
```

    - **Evaluate Model**

```
SELECT
     *
   FROM
     ML.EVALUATE(MODEL `my_dataset.purchase_classification_model`)
```

    - **Predict Using Model**

```
SELECT
     customer_id,
     predicted_purchase
   FROM
     ML.PREDICT(MODEL `my_dataset.purchase_classification_model`,
       (
         SELECT
           customer_id,
           age,
           income,
           browsing_time
         FROM
           `my_dataset.new_customer_data`
       ));
```

  - **Regression**

- **Create Model**

```
CREATE OR REPLACE MODEL `my_dataset.house_price_regression_model`
OPTIONS(
  model_type='linear_reg',
  input_label_cols=['price']
) AS
SELECT
  square_footage,
  num_bedrooms,
  num_bathrooms,
  location,
  price
FROM
  `my_dataset.house_data`;
```

- **Evaluate**

```
SELECT
  *
FROM
  ML.EVALUATE(MODEL `my_dataset.house_price_regression_model`)
```

- **Predict**

```
  SELECT
    house_id,
    predicted_price
  FROM
    ML.PREDICT(MODEL `my_dataset.house_price_regression_model`,
      (
        SELECT
          house_id,
          square_footage,
          num_bedrooms,
          num_bathrooms,
          location
        FROM
          `my_dataset.new_house_data`
      ));
```

  - **Forecasting**
  - **Clustering**
  - **List all the models**

```
  SELECT
    model_id,
    model_type,
    creation_time,
    last_modified_time
  FROM
    ML.MODELS();
```

  - **Deleting Model**

```
  DROP MODEL `my_dataset.my_model`;
```

- **Other features:**

- Dataform: It helps to orchestrate and manage data pipelines, transformations, and workflows in a more structured and reproducible manner.
- Orchestration
- SQL translation
- Monitoring

Container Registry:

Google Cloud Container Registry (GCR) is a fully managed Docker container registry that makes it easy for developers to store, manage, and deploy Docker container images. It integrates seamlessly with Google Cloud Platform (GCP) services and tools, providing a secure and scalable solution for storing and distributing Docker images.

Artifact Registry: