```python
# Fill in student ID and name
#
student_id = ""
student_first_last_name = ""
print(student_id, student_first_last_name)

"""
     Firebase Realtime database demonstration.

    @Ahsan Habib
    School of IT,
    Deakin University, Australia.
"""

# Install libraris, if not yet.
! pip install firebase_admin pandas

import firebase_admin

databaseURL = '<your database URL>'
cred_obj = firebase_admin.credentials.Certificate(
    '<your certificate json file>.json'
)
default_app = firebase_admin.initialize_app(cred_obj, {
     'databaseURL':databaseURL
     })

from firebase_admin import db

# A reference point is always needed to be set
# before any operation is carried out on a database.
#
ref = db.reference("/")

# JSON format data (key/value pair)
data = {  # Outer {} contains inner data structure
     "Book1":
     {
          "Title": "The Fellowship of the Ring",
          "Author": "J.R.R. Tolkien",
          "Genre": "Epic fantasy",
          "Price": 100
     },
     "Book2":
     {
          "Title": "The Two Towers",
          "Author": "J.R.R. Tolkien",
          "Genre": "Epic fantasy",
          "Price": 100
     },
```

```
      "Book3":
      {
            "Title": "The Return of the King",
            "Author": "J.R.R. Tolkien",
            "Genre": "Epic fantasy",
            "Price": 100
      },
      "Book4":
      {
            "Title": "Brida",
            "Author": "Paulo Coelho",
            "Genre": "Fiction",
            "Price": 100
      }
}

# JSON format data is set (overwritten) to the reference
# point set at /, which is the root node.
#
ref.set(data)

ref = db.reference("/")  # set ref point

# query all data under the ref
books = ref.get()
print(books)
print(type(books))

# print each item separately
for key, value in books.items():
    print(f"{key}: {value}")


# Query /Book1
ref = db.reference("/Book1")
books = ref.get()
print(books)

# Write using push() function
# Note that a set() is called on top of push()
#
ref = db.reference("/")
ref.set({
      "Books":
      {
            "Best_Sellers": -1
      }
})

ref = db.reference("/Books/Best_Sellers")
```

```
for key, value in data.items():
     ref.push().set(value)

# Update data
#
# Requirement: The price of the books by
# J. R. R. Tolkien is reduced to 80 units to
# offer a discount.
#
ref = db.reference("/Books/Best_Sellers/")
best_sellers = ref.get()
print(best_sellers)
for key, value in best_sellers.items():
     if(value["Author"] == "J.R.R. Tolkien"):
          value["Price"] = 90
          ref.child(key).update({"Price":80})

# Let's delete all best seller books
# with J.R.R. Tolkien as the author.
#
ref = db.reference("/Books/Best_Sellers")

for key, value in best_sellers.items():
     if(value["Author"] == "J.R.R. Tolkien"):
          ref.child(key).set({})

# Delete all best_seller data.
#
ref = db.reference("/Books/Best_Sellers/")
best_sellers = ref.get()
print(best_sellers)
print(type(best_sellers))

ref = db.reference("/Books/Best_Sellers")
ref.set({})
```