|  |  |  |
| --- | --- | --- |
| UniqueRef | Name | Phone |
| dsfgjtgrtr | Yewande | 07037953367 |
| Kjkdjgadjks | Segun | 07037953357 |
| Lkdlkhsjdkjs | Kevin | 07087953367 |
| Dklskhghjjd | Yewande | 07037953367 |
| ,mkjgfyhjlds | Segun | 07037953357 |
| Iuyugbjkdslds | Kevin | 07087953367 |
| Kjhjhgfvhjknlm | Yewande | 07037953678 |
| Dsjchgusyiusokd; | Segun | 07037953357 |
| Lkjigdbvjkldl | Kevin | 07087953367 |
| Dlkjihu | Yewande | 07037953367 |
| Sdkjiu | Segun | 07037953357 |
| Ewr | Kevin | 07087953367 |
| sfdgdsdsdf | Stella | 08033765423 |

Table name => Student table

Scene 1

Y => Give me the phone number on second row

S => Give me the second phone number on the table

Scene 2

Y =>

S => Fetch me the phone number with the id **Kjhjhgfvhjknlm**

**Fetch me the phone number from the student table where the unique ref is** Kjhjhgfvhjknlm

C => Create -- INSERT

R => Read -- SELECT

U => Update -- UPDATE

D => Delete – DELETE

Syntax =>

CREATE => INSERT INTO {TABLE NAME} ({column1} .. {column n}) VALUES ({value 1} .. {value n});

e.g INSERT INTO Student(UniqueRef, Name, Phone) VALUES(sfdgdsdsdf, Stella, 08033765423);

READ => SELECT {parameter/ all(\*)} FROM {TABLE NAME};

e.g SELECT \* FROM Student

UPDATE => UPDATE {TABLE NAME} SET {COLUMN NAME} = {VALUE};

e.g UPDATE Student SET name = Ayotola

Tables are made of rows and columns

A table must have at least:

1 Row

1 Column

|  |
| --- |
|  |

SQL => Sequel, SQL, Full meaning => Structured Query Language