# Python II

HIGHER DIPLOMA IN DATA ANALYTICS

# PyMySQL

# PyMySQL

- MySQLdb
- mysql.connector
- PyMySQL

# connect()

- The connect() function connects to a MySQL database

# connect()

- The connect() function connects to a MySQL database
- **host** - Host where the database server is located

# connect()

- The connect() function connects to a MySQL database

- **host** - Host where the database server is located
- **user** - Username to log in as

# connect()

- The connect() function connects to a MySQL database

- **host** - Host where the database server is located
- **user** - Username to log in as
- **password** - Password to use

# connect()

- The connect() function connects to a MySQL database

- **host** - Host where the database server is located
- **user** - Username to log in as
- **password** - Password to use
- **db** - Database to use

# connect()

- The connect() function connects to a MySQL database

- **host** - Host where the database server is located
- **user** - Username to log in as
- **password** - Password to use
- **db** - Database to use
- **port** - Port to use

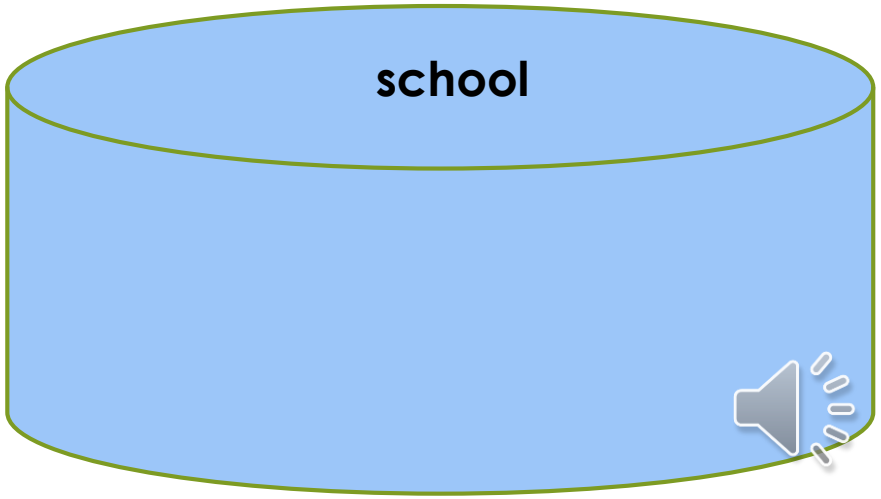
# connect()

- The connect() function connects to a MySQL database

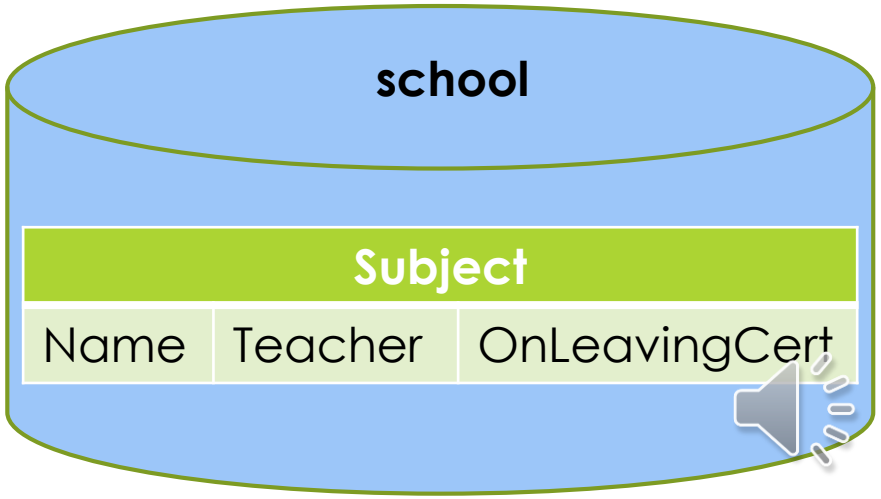
- **host** - Host where the database server is located
- **user** - Username to log in as
- **password** - Password to use
- **db** - Database to use
- **port** - Port to use
- **cursorclass** - Custom cursor class to use

# Connecting to the Database

# Connecting to the Database

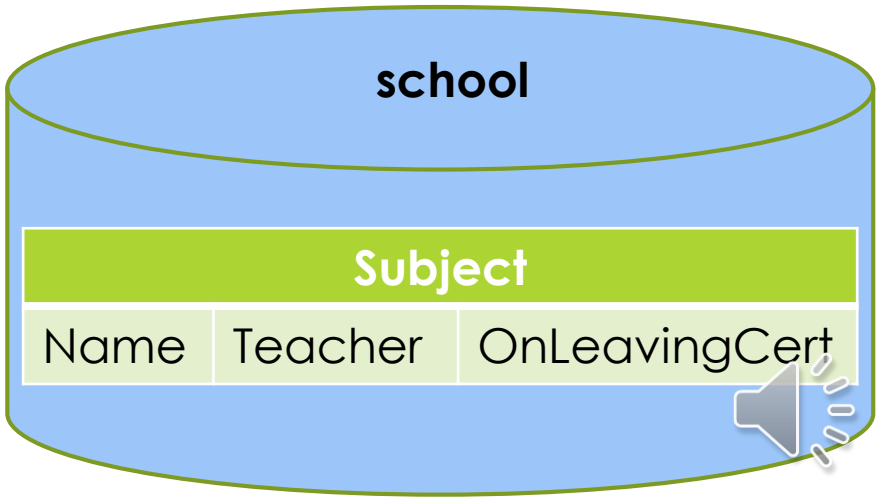


# Connecting to the Database



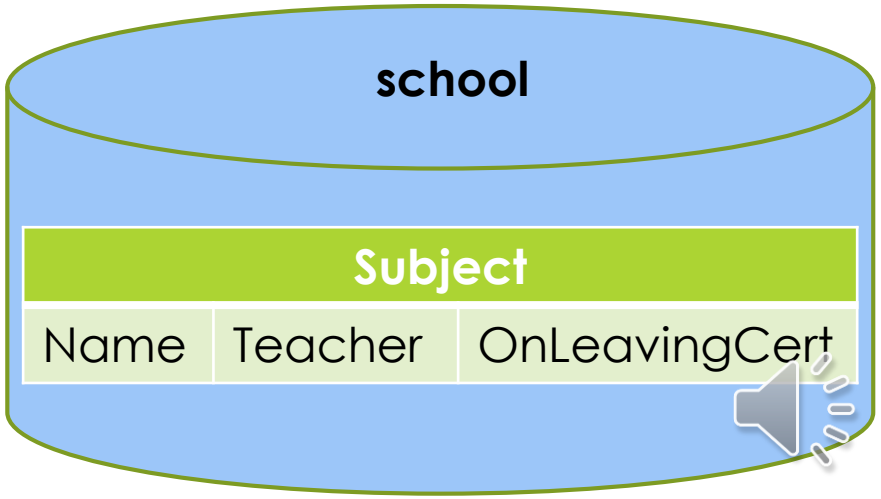
# Connecting to the Database

```
pymysql.connect(
```



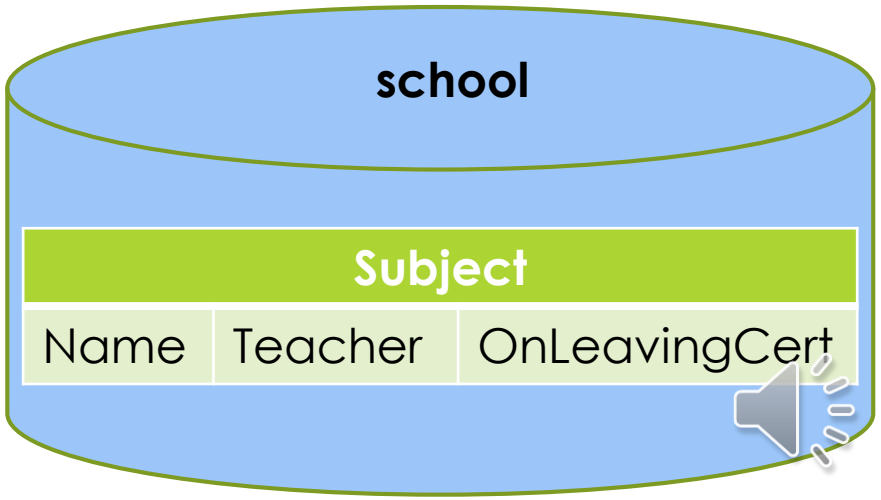
# Connecting to the Database

```
pymysql.connect( "localhost",
```



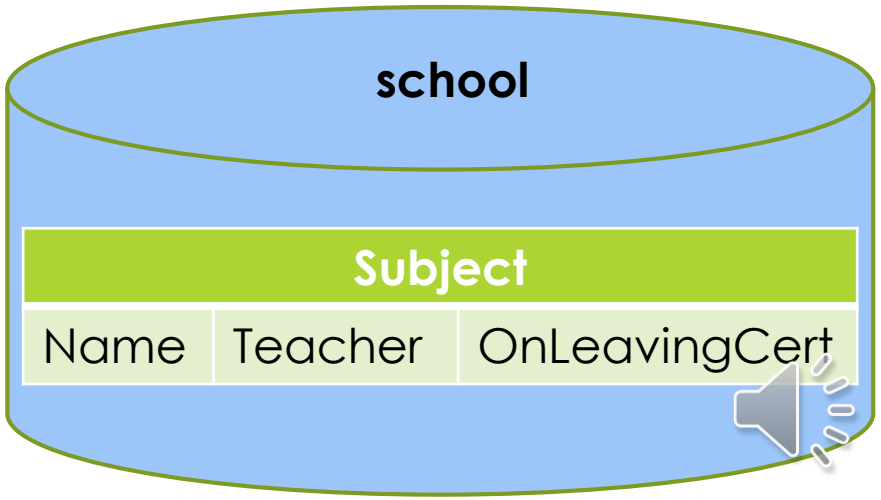
# Connecting to the Database

```
pymysql.connect( "localhost",  "root",
```



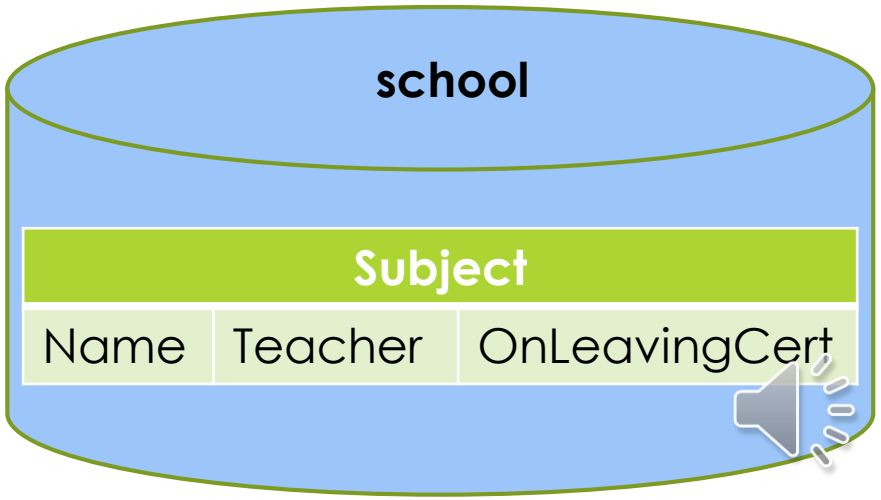
# Connecting to the Database

```
pymysql.connect( "localhost",  "root",   "root",
```



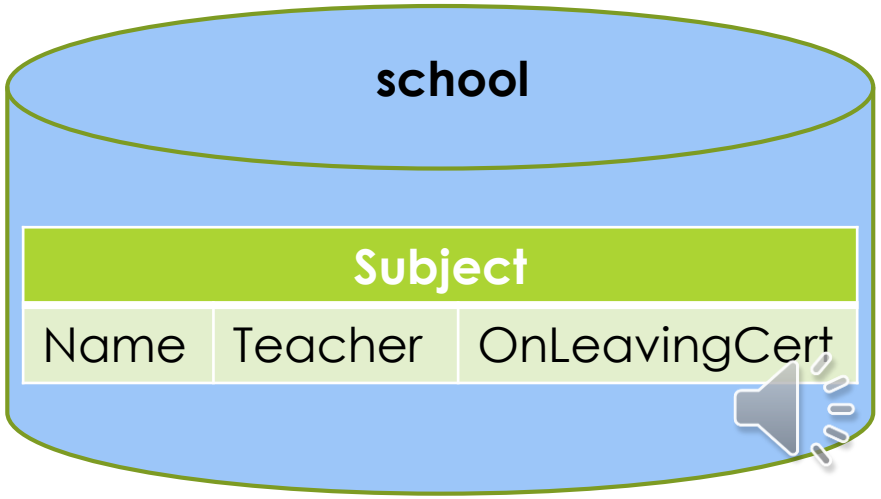
# Connecting to the Database

```
pymysql.connect( "localhost", "root",  "root",  "school",
```



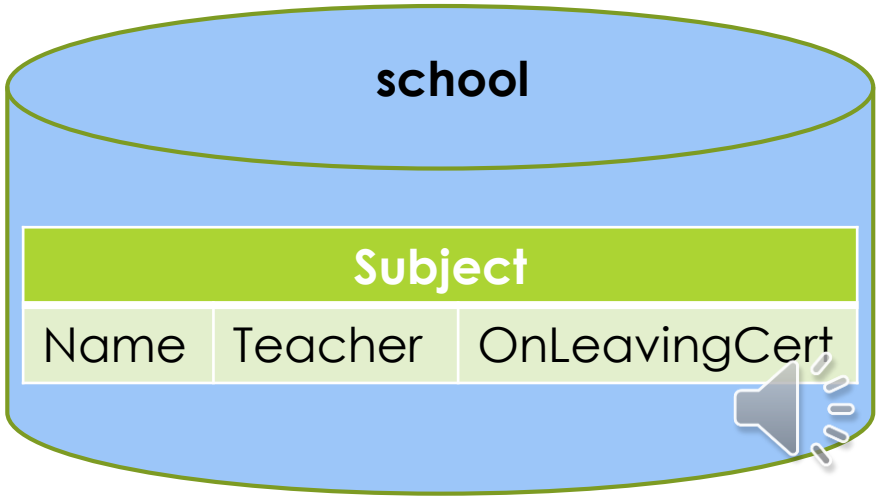
# Connecting to the Database

```
pymysql.connect( "localhost", "root",  "root",  "school",
                 cursorclass=pymysql.cursors.DictCursor)
```



# Connecting to the Database

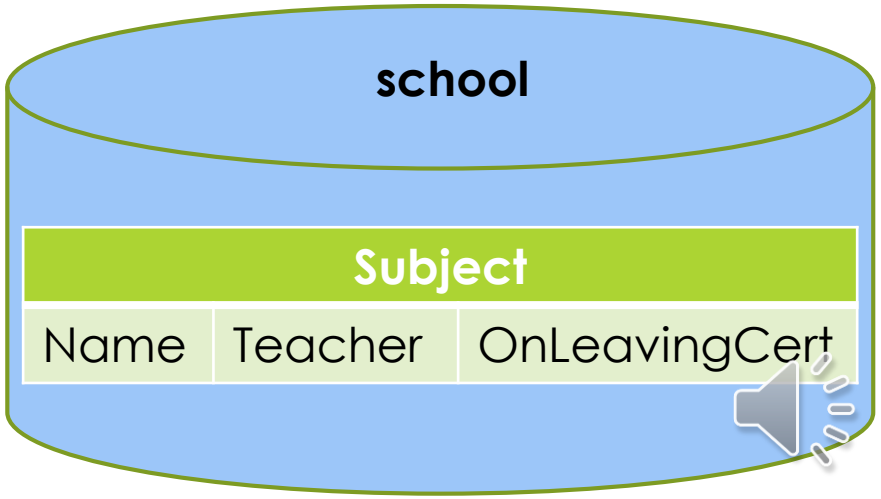
```
conn = pymysql.connect( "localhost", "root",  "root",  "school",
                        cursorclass=pymysql.cursors.DictCursor)
```



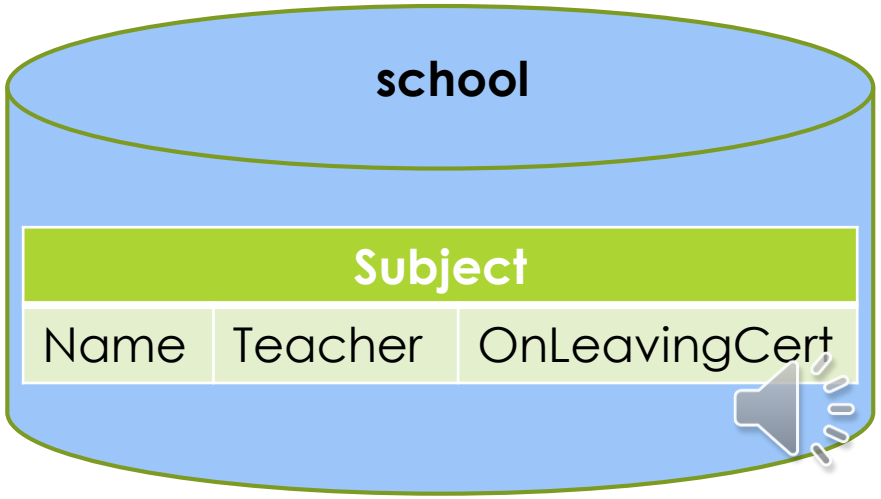
# Connecting to the Database

```
conn = pymysql.connect( "localhost", "root",  "root",  "school",
                        cursorclass=pymysql.cursors.DictCursor)
```

```
conn = pymysql.connect(user="root",
                       cursorclass=pymysql.cursors.DictCursor,
                       password="root",
                       host="localhost",
                       db="school",
                       port=3306)
```

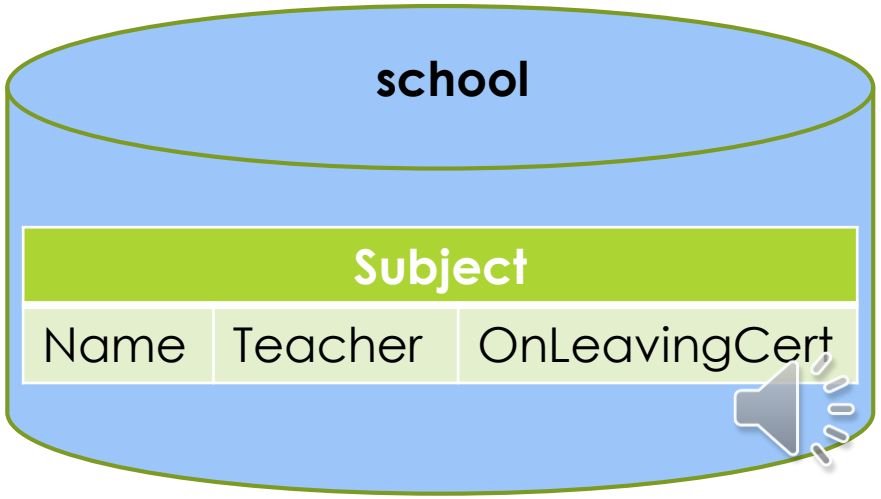


# Executing a query



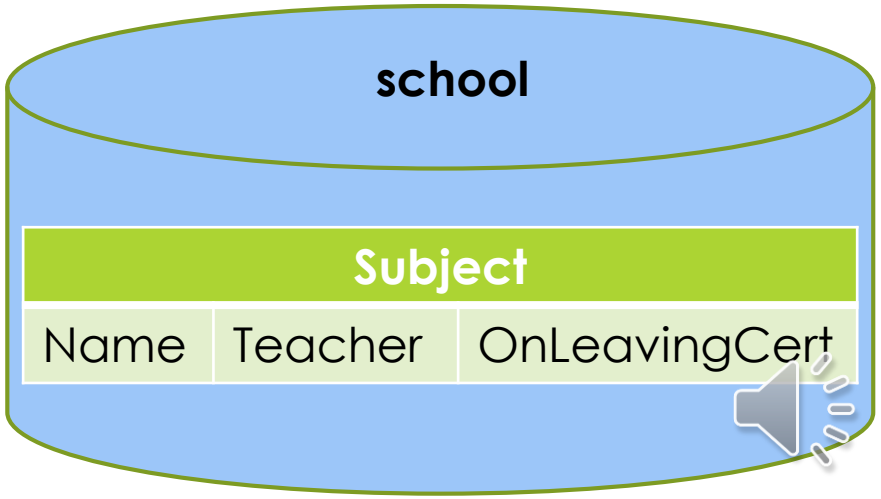
# Executing a query

```
query = "SELECT * FROM subject"
```



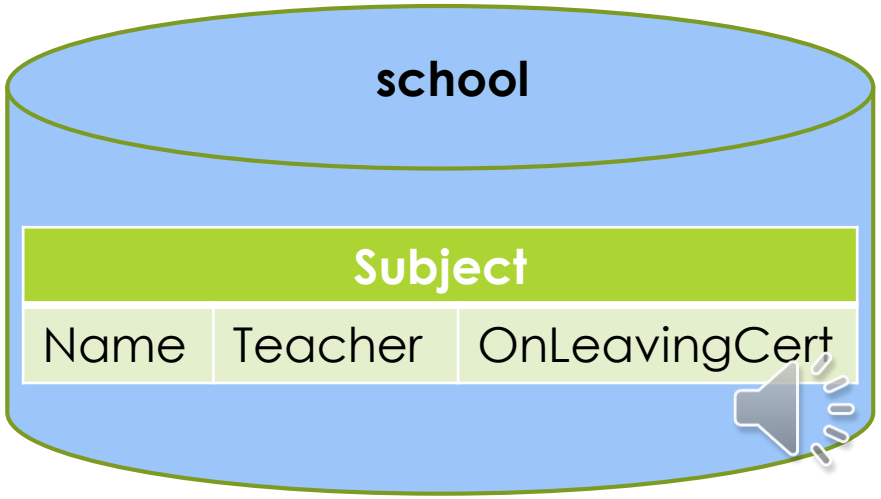
# Executing a query

```
query = "SELECT * FROM subject"

with conn:
```



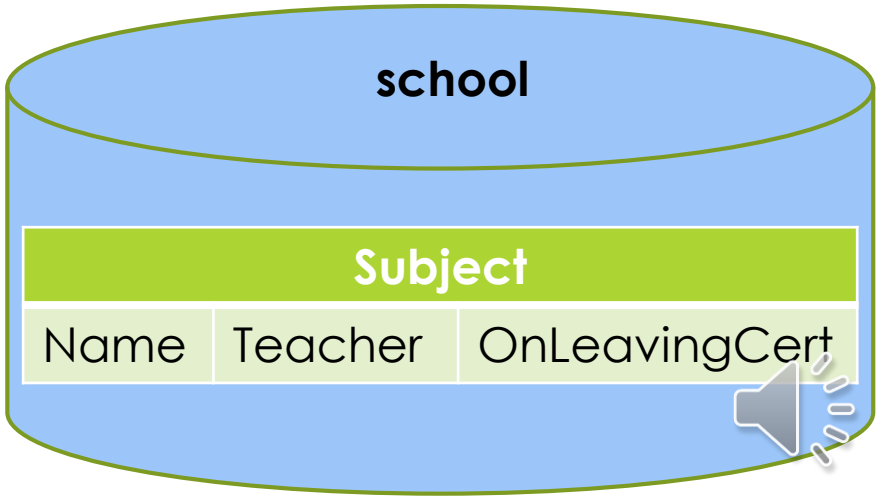
# Executing a query

```
query = "SELECT * FROM subject"

with conn:
    cursor = conn.cursor()
```



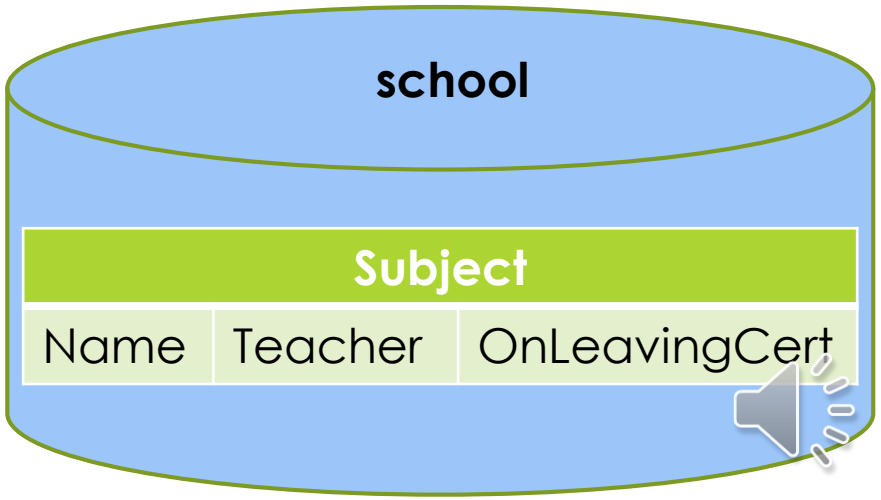
# Executing a query

```
query = "SELECT * FROM subject"

with conn:
    cursor = conn.cursor()
    cursor.execute(query)
```



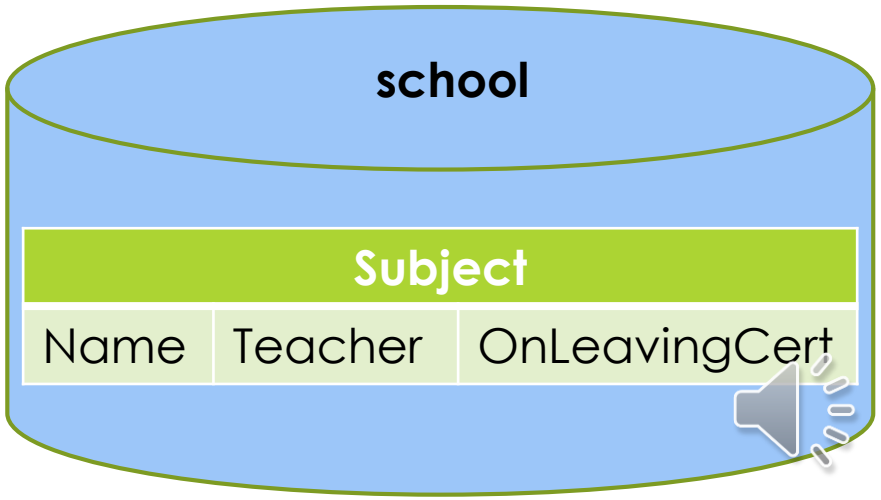
# Executing a query

```
query = "SELECT * FROM subject"

with conn:
    cursor = conn.cursor()
    cursor.execute(query)
    subjects = cursor.fetchall()
```



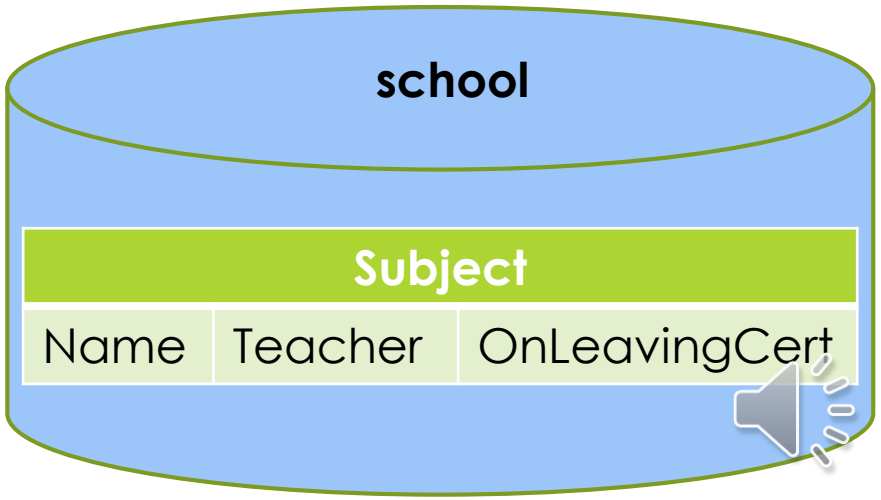
# Executing a query

```
query = "SELECT * FROM subject"

with conn:
    cursor = conn.cursor()
    cursor.execute(query)
    subjects = cursor.fetchall()
    for s in subjects:
```



# Executing a query

```
query = "SELECT * FROM subject"

with conn:
    cursor = conn.cursor()
    cursor.execute(query)
    subjects = cursor.fetchall()
    for s in subjects:
        print(s["Name"])
```



# Executing a query

# Executing a query

**school**

| Subject | | |
|---|---|---|
| **Name** | **Teacher** | **OnLeavingCert** |
| Biology | Mr. Pasteur | 1 |
| Colouring | Mr. Picasso | 0 |
| English | Mr. Kavanagh | 1 |
| French | Ms. Dubois | 1 |
| Maths | Mr. Hawking | 1 |
| Religion | Fr. Lynch | 1 |
| Spelling | Ms. Smith | 0 |

# Executing a query

```
query = "SELECT * FROM subject
         WHERE teacher LIKE %s"
```

**school**

**Subject**

| Name | Teacher | OnLeavingCert |
|---|---|---|
| Biology | Mr. Pasteur | 1 |
| Colouring | Mr. Picasso | 0 |
| English | Mr. Kavanagh | 1 |
| French | Ms. Dubois | 1 |
| Maths | Mr. Hawking | 1 |
| Religion | Fr. Lynch | 1 |
| Spelling | Ms. Smith | 0 |

# Executing a query

```
query = "SELECT * FROM subject
         WHERE teacher LIKE %s"

with conn:
```

**school**

| Subject | | |
|---|---|---|
| **Name** | **Teacher** | **OnLeavingCert** |
| Biology | Mr. Pasteur | 1 |
| Colouring | Mr. Picasso | 0 |
| English | Mr. Kavanagh | 1 |
| French | Ms. Dubois | 1 |
| Maths | Mr. Hawking | 1 |
| Religion | Fr. Lynch | 1 |
| Spelling | Ms. Smith | 0 |

# Executing a query

```
query = "SELECT * FROM subject
         WHERE teacher LIKE %s"

with conn:
    cursor = conn.cursor()
```

**school**

| Subject | | |
|---|---|---|
| **Name** | **Teacher** | **OnLeavingCert** |
| Biology | Mr. Pasteur | 1 |
| Colouring | Mr. Picasso | 0 |
| English | Mr. Kavanagh | 1 |
| French | Ms. Dubois | 1 |
| Maths | Mr. Hawking | 1 |
| Religion | Fr. Lynch | 1 |
| Spelling | Ms. Smith | 0 |

# Executing a query

```
query = "SELECT * FROM subject
         WHERE teacher LIKE %s"

with conn:
    cursor = conn.cursor()
    cursor.execute(query, ("Ms.%"))
```

**school**

**Subject**

| Name | Teacher | OnLeavingCert |
|---|---|---|
| Biology | Mr. Pasteur | 1 |
| Colouring | Mr. Picasso | 0 |
| English | Mr. Kavanagh | 1 |
| French | Ms. Dubois | 1 |
| Maths | Mr. Hawking | 1 |
| Religion | Fr. Lynch | 1 |
| Spelling | Ms. Smith | 0 |

# Executing a query

```
query = "SELECT * FROM subject
         WHERE teacher LIKE %s"

with conn:
    cursor = conn.cursor()
    cursor.execute(query, ("Ms.%"))
    subjects = cursor.fetchall()
```

**school**

**Subject**

| Name | Teacher | OnLeavingCert |
|---|---|---|
| Biology | Mr. Pasteur | 1 |
| Colouring | Mr. Picasso | 0 |
| English | Mr. Kavanagh | 1 |
| French | Ms. Dubois | 1 |
| Maths | Mr. Hawking | 1 |
| Religion | Fr. Lynch | 1 |
| Spelling | Ms. Smith | 0 |

# Executing a query

```
query = "SELECT * FROM subject
         WHERE teacher LIKE %s"

with conn:
    cursor = conn.cursor()
    cursor.execute(query, ("Ms.%"))
    subjects = cursor.fetchall()
    for s in subjects:
```

**school**

**Subject**

| Name | Teacher | OnLeavingCert |
|---|---|---|
| Biology | Mr. Pasteur | 1 |
| Colouring | Mr. Picasso | 0 |
| English | Mr. Kavanagh | 1 |
| French | Ms. Dubois | 1 |
| Maths | Mr. Hawking | 1 |
| Religion | Fr. Lynch | 1 |
| Spelling | Ms. Smith | 0 |

# Executing a query

```
query = "SELECT * FROM subject
         WHERE teacher LIKE %s"

with conn:
    cursor = conn.cursor()
    cursor.execute(query, ("Ms.%"))
    subjects = cursor.fetchall()
    for s in subjects:
        print(s["Name"])
```

**school**

**Subject**

| Name | Teacher | OnLeavingCert |
|---|---|---|
| Biology | Mr. Pasteur | 1 |
| Colouring | Mr. Picasso | 0 |
| English | Mr. Kavanagh | 1 |
| French | Ms. Dubois | 1 |
| Maths | Mr. Hawking | 1 |
| Religion | Fr. Lynch | 1 |
| Spelling | Ms. Smith | 0 |

# Executing a query

```
query = "SELECT * FROM subject
         WHERE teacher LIKE %s"

with conn:
    cursor = conn.cursor()
    cursor.execute(query, ("Ms.%"))
    subjects = cursor.fetchall()
    for s in subjects:
        print(s["Name"])
```

French
Spelling

**school**

**Subject**

| Name | Teacher | OnLeavingCert |
|---|---|---|
| Biology | Mr. Pasteur | 1 |
| Colouring | Mr. Picasso | 0 |
| English | Mr. Kavanagh | 1 |
| French | Ms. Dubois | 1 |
| Maths | Mr. Hawking | 1 |
| Religion | Fr. Lynch | 1 |
| Spelling | Ms. Smith | 0 |

# Inserting new data

**school**

| Subject | | |
|---|---|---|
| **Name*** | **Teacher** | **OnLeavingCert** |
| Biology | Mr. Pasteur | 1 |
| Colouring | Mr. Picasso | 0 |
| English | Mr. Kavanagh | 1 |
| French | Ms. Dubois | 1 |
| Maths | Mr. Hawking | 1 |
| Religion | Fr. Lynch | 1 |
| Spelling | Ms. Smith | 0 |

# Inserting new data

```
ins = "INSERT INTO subject
        (Name, Teacher, OnLeavingCert)
        VALUES (%s, %s, %s)"
```

**school**

| Subject | | |
|---|---|---|
| **Name*** | **Teacher** | **OnLeavingCert** |
| Biology | Mr. Pasteur | 1 |
| Colouring | Mr. Picasso | 0 |
| English | Mr. Kavanagh | 1 |
| French | Ms. Dubois | 1 |
| Maths | Mr. Hawking | 1 |
| Religion | Fr. Lynch | 1 |
| Spelling | Ms. Smith | 0 |

# Inserting new data

```
ins = "INSERT INTO subject
        (Name, Teacher, OnLeavingCert)
        VALUES (%s, %s, %s)"
with conn:
    cursor = conn.cursor()
```

**school**

| Subject | | |
|---|---|---|
| **Name*** | **Teacher** | **OnLeavingCert** |
| Biology | Mr. Pasteur | 1 |
| Colouring | Mr. Picasso | 0 |
| English | Mr. Kavanagh | 1 |
| French | Ms. Dubois | 1 |
| Maths | Mr. Hawking | 1 |
| Religion | Fr. Lynch | 1 |
| Spelling | Ms. Smith | 0 |

# Inserting new data

```
ins = "INSERT INTO subject
        (Name, Teacher, OnLeavingCert)
        VALUES (%s, %s, %s)"
with conn:
    cursor = conn.cursor()
    cursor.execute(ins, ("Maths", "Ms. Jones", 1))
```

**school**

**Subject**

| Name* | Teacher | OnLeavingCert |
|---|---|---|
| Biology | Mr. Pasteur | 1 |
| Colouring | Mr. Picasso | 0 |
| English | Mr. Kavanagh | 1 |
| French | Ms. Dubois | 1 |
| Maths | Mr. Hawking | 1 |
| Religion | Fr. Lynch | 1 |
| Spelling | Ms. Smith | 0 |

# Inserting new data

```
ins = "INSERT INTO subject
        (Name, Teacher, OnLeavingCert)
        VALUES (%s, %s, %s)"
with conn:
    cursor = conn.cursor()
    cursor.execute(ins, ("Maths", "Ms. Jones", 1))
    conn.commit()
```

**school**

**Subject**

| Name* | Teacher | OnLeavingCert |
|---|---|---|
| Biology | Mr. Pasteur | 1 |
| Colouring | Mr. Picasso | 0 |
| English | Mr. Kavanagh | 1 |
| French | Ms. Dubois | 1 |
| Maths | Mr. Hawking | 1 |
| Religion | Fr. Lynch | 1 |
| Spelling | Ms. Smith | 0 |

# Inserting new data

```
ins = "INSERT INTO subject
       (Name, Teacher, OnLeavingCert)
       VALUES (%s, %s, %s)"
with conn:
    try:
        cursor = conn.cursor()
        cursor.execute(query,
                       ("Maths", "Ms. Jones", "1"))
        conn.commit()
        print("Insert Successful")
    except:
        print("Maths already exists")
```

**school**

| Subject | | |
|---|---|---|
| **Name*** | **Teacher** | **OnLeavingCert** |
| Biology | Mr. Pasteur | 1 |
| Colouring | Mr. Picasso | 0 |
| English | Mr. Kavanagh | 1 |
| French | Ms. Dubois | 1 |
| Maths | Mr. Hawking | 1 |
| Religion | Fr. Lynch | 1 |
| Spelling | Ms. Smith | 0 |

# Exceptions

# Exceptions

```
name = "Maths"
teacher = "Ms. Jones"
lc = 1
with conn:
    try:
        cursor = conn.cursor()
        cursor.execute(query, (name, teacher, lc))
        conn.commit()
        print("Insert Successful")
    except pymysql.err.InternalError as e:
        print("Internal Error", e)
    except pymysql.err.IntegrityError:
        print("Error", name, "already exists")
    except Exception as e:
        print("error", e)
```

# Exceptions

```
name = "Maths"
teacher = "Ms. Jones"
lc = 1
with conn:
    try:
        cursor = conn.cursor()
        cursor.execute(query, (name, teacher, lc))
        conn.commit()
        print("Insert Successful")
    except pymysql.err.InternalError as e:
        print("Internal Error", e)
    except pymysql.err.IntegrityError:
        print("Error", name, "already exists")
    except Exception as e:
        print("error", e)
```

# Exceptions

```
name = "Maths"
teacher = "Ms. Jones"
lc = 1
with conn:
    try:
        cursor = conn.cursor()
        cursor.execute(query, (name, teacher, lc))
        conn.commit()
        print("Insert Successful")
    except pymysql.err.InternalError as e:
        print("Internal Error", e)
    except pymysql.err.IntegrityError:
        print("Error", name, "already exists")
    except Exception as e:
        print("error", e)
```

```
name = "Spanish"
teacher = "Ms. Jones"
lc = "yes"
with conn:
    try:
        cursor = conn.cursor()
        cursor.execute(query, (name, teacher, lc))
        conn.commit()
        print("Insert Successful")
    except pymysql.err.InternalError as e:
        print("Internal Error", e)
    except pymysql.err.IntegrityError:
        print("Error", name, "already exists")
    except Exception as e:
        print("error", e)
```

# Exceptions

```
name = "Maths"
teacher = "Ms. Jones"
lc = 1
with conn:
    try:
        cursor = conn.cursor()
        cursor.execute(query, (name, teacher, lc))
        conn.commit()
        print("Insert Successful")
    except pymysql.err.InternalError as e:
        print("Internal Error", e)
    except pymysql.err.IntegrityError:
        print("Error", name, "already exists")
    except Exception as e:
        print("error", e)
```

```
name = "Spanish"
teacher = "Ms. Jones"
lc = "yes"
with conn:
    try:
        cursor = conn.cursor()
        cursor.execute(query, (name, teacher, lc))
        conn.commit()
        print("Insert Successful")
    except pymysql.err.InternalError as e:
        print("Internal Error", e)
    except pymysql.err.IntegrityError:
        print("Error", name, "already exists")
    except Exception as e:
        print("error", e)
```

# Deleting data

```
query = "DELETE FROM subject WHERE NAME = %s"

name = "Maths"

with conn:
    try:
        cursor = conn.cursor()
        rowsAffected = cursor.execute(query, (name))
        conn.commit()
        if (rowsAffected == 0):
            print("Nothing deleted - ", name, "never existed")
        else:
            print(rowsAffected, "row(s) deleted")
    except  Exception as e:
        print("error", e)
```

# Updating data

```
query = "UPDATE subject SET teacher = %s WHERE name = %s"
subject = "Maths"
newTeacher = "Mr. Murphy"

with conn:
    try:
        cursor = conn.cursor()
        rowsAffected = cursor.execute(query, (newTeacher, subject))
        conn.commit()
        if (rowsAffected == 0):
            print(subject, "not updated")
        else:
            print(subject, "now taught by", newTeacher)
    except  Exception as e:
        print("error", e)
```