In [14]:
```
# Import date from datetime
import datetime

# Create a date object
my_birth_date = date(2002, 10, 26)

# Which day of the week is the date?
print(my_birth_date.weekday(),'Saturday')
```

```
5 Saturday
```

In [9]:
```
today=date.today()
# Subtract the two dates and print the number of days
print((today - my_birth_date ).days)
```

```
7613
```

ISO Format

In [11]:
```
my_birth_date.isoformat()
```

Out[11]:
```
'2002-10-26'
```

STRFTIME

In [12]:
```
my_birth_date.strftime("%m/%d/%Y")
```

Out[12]:
```
'10/26/2002'
```

Creating datetimes by hand

In [29]:
```
import datetime

dt = datetime.datetime(2017, 12, 31, 15, 19, 13)

print(dt.isoformat())
```

```
2017-12-31T15:19:13
```

Turning strings into datetimes

In [36]:
```
# Starting string, in YYYY-MM-DD HH:MM:SS format
s = '2017-02-03 00:00:01'

# Write a format string to parse s
fmt = '%Y-%m-%d %H:%M:%S'
nrm= '%Y-%m-%d %H:%M:%S'

# Create a datetime object d
d = datetime.datetime.strptime(s, fmt)
print(d)
```

```
2017-02-03 00:00:01
```

Creating timezone aware datetimes

In [37]:
```
# Import datetime, timezone
from datetime import datetime, timezone

# October 1, 2017 at 15:26:26, UTC
dt = datetime(2017, 10, 1, 15, 26, 26, tzinfo=timezone.utc)

# Print results
print(dt.isoformat())
```

```
2017-10-01T15:26:26+00:00
```

In [43]:
```
# Import datetime, timedelta, timezone
from datetime import datetime, timedelta, timezone

# Create a timezone for Pakistan Standard Time, or UTC-8
pst = timezone(timedelta(hours=-5))

# October 1, 2017 at 15:26:26, UTC-8
dt = datetime(2023, 10, 26, 8, 26, 26, tzinfo=pst)

# Print results
print(dt.isoformat())
```

```
2023-10-26T08:26:26-05:00
```

In [47]:

```
# Import tz
from dateutil import tz

pt= tz.gettz('Asia\Karachi')

# October 1, 2017 at 15:26:26, UTC-8
dt = datetime(2023, 10, 26, 8, 26, 26, tzinfo=pt)

# Print results
print(dt.isoformat())
```

```
2023-10-26T08:26:26
```