Date time module

- we can import datatime by
    - import datetime

1. formats of datetime

- 10-01-20 - %d %m %y
- 10 Jan 2020 - %d %b %Y
- 2020-01-10 - %Y-%m-%d
- 10/01/2020 - %d/%m/%Y

commands

1. current_datetime = datetime.datetime().now()

- current_datetime.now()
- current_datetime.today()
- current_datetime.weekday()
- current_datetime.date()
- current_datetime.day
- current_datetime.year
- current_datetime.month

2. Timedelta

```
A timedelta object represents a duration, the difference between two dates or times.

```

```
class datetime.timedelta(days=0, seconds=0, microseconds=0, milliseconds=0, minutes=0, hours=0, weeks=0)

All arguments are optional and default to 0. Arguments may be integers or floats, and may be positive or negative.

- syntax-
        - current_day = current_date + datetime.timedelta(days,weeks,hours..=?)

### NOTE:- in this module we can only play with days,weeks,hours=1,2,3,4,...
```

In [ ]:

```

```

3. strftime()

> - syntax :-
>
> data.strftime(format)

- string from time object(time to string)
- The strftime() function is used to convert date and time objects to their string representation. It takes one or more input of formatted code and returns the string representation.

```
The table below shows all the codes that you can pass to the strftime() method.

Directive   Meaning Example
- %a    Abbreviated weekday name.   Sun, Mon, ...

- %A    Full weekday name.  Sunday, Monday, ...

- %w    Weekday as a decimal number.    0, 1, ..., 6

- %d    Day of the month as a zero-padded decimal.  01, 02, ..., 31

- %b    Abbreviated month name. Jan, Feb, ..., Dec

- %B    Full month name.    January, February, ...

- %m    Month as a zero-padded decimal number.  01, 02, ..., 12

- %y    Year without century as a zero-padded decimal number.   00, 01, ..., 99

- %-y   Year without century as a decimal number.   0, 1, ..., 99

- %Y    Year with century as a decimal number.  2013, 2019 etc.

- %H    Hour (24-hour clock) as a zero-padded decimal number.   00, 01, ..., 23

- %I    Hour (12-hour clock) as a zero-padded decimal number.   01, 02, ..., 12

- %p    Locale’s AM or PM.  AM, PM

- %M    Minute as a zero-padded decimal number. 00, 01, ..., 59

- %S    Second as a zero-padded decimal number. 00, 01, ..., 59

- %f    Microsecond as a decimal number, zero-padded on the left.   000000 - 999999

- %z    UTC offset in the form +HHMM or -HHMM.

- %Z    Time zone name.

- %j    Day of the year as a zero-padded decimal number.    001, 002, ..., 366

- %U    Week number of the year (Sunday as the first day of the week). All days in a new year
preceding         the first Sunday are considered to be in week 0.    00, 01, ..., 53

- %W    Week number of the year (Monday as the first day of the week). All days in a new year
preceding         the first Monday are considered to be in week 0.    00, 01, ..., 53

- %c    Locale’s appropriate date and time representation.  Mon Sep 30 07:06:05 2013

- %x    Locale’s appropriate date representation.   09/30/13

- %X    Locale’s appropriate time representation.   07:06:05

- %%    A literal '%' character.    %
```

In [ ]:

4. strptime()

- string point to time(string to point objects)

- syntax -

> ```
>  - date = "20/02/2022"
> - date_format = "%d/%m/%Y"
>
> - date = datetime.datetime.strptime(date, date_format)
> ```

- date = 11/20/2022
- date_format = %d/%m/%Y

NOTE:- style of date and dateformat shuould be match

In [ ]: