python_时间戳处理

```
def is_weekend(date):
    return date.strftime("%w") == "0" or date.strftime("%w") == "6"



def getDays(num, weekend):
    days = []
    for i in range(1, num * 7):
        day = datetime.datetime.now() + datetime.timedelta(days=-i)
        if is_weekend(day) and weekend:
            days.append(day.strftime('%Y%m%d'))
        elif not is_weekend(day) and not weekend:
            days.append(day.strftime('%Y%m%d'))
        if len(days) >= num:
            break
    return days



def get_ts(num):
    return int(time.mktime(datetime.date.today().timetuple()) - num * 60 * 60 * 24) * 1000



def get_ds(num):
    return (datetime.datetime.now() + datetime.timedelta(days=-num)).strftime('%Y%m%d')



def get_ds_hour(days=0, hours=0):
    return (datetime.datetime.now() + datetime.timedelta(days=-days, hours=hours)).strftime('%Y%m%d%H')



def get_date(num, format='%Y-%m-%d'):
    return (datetime.datetime.now() + datetime.timedelta(days=-num)).strftime(format)



def get_month_first(num):
    date = (datetime.date.today() - relativedelta(months=num))
```

```
    return datetime.date(date.year, date.month, 1)



def get_month_first_date(num):

    return get_month_first(num).strftime('%Y-%m-%d')



def get_month_first_ds(num):

    return get_month_first(num).strftime('%Y%m%d')



def get_month_first_ts(num):

    return date2ts(get_month_first(num))



def get_month(num):

    return get_month_first(num).strftime('%Y%m')



def get_week_first(num):

    now = datetime.date.today()

    return now - datetime.timedelta(days=now.weekday()) - relativedelta(weeks=num)



def get_week_last(num):

    now = datetime.date.today()

    return now + datetime.timedelta(days=6 - now.weekday()) - relativedelta(weeks=num)



def get_week_first_date(num):

    return get_week_first(num).strftime('%Y-%m-%d')



def get_week_first_ds(num):

    return get_week_first(num).strftime('%Y%m%d')
```

```
def get_week_first_ts(num):
    return date2ts(get_week_first(num))



def get_week_last_ds(num):
    return get_week_last(num).strftime('%Y%m%d')



def ts2date(ts, format="%Y-%m-%d %H:%M:%S"):
    return time.strftime(format, time.localtime(ts / 1000))



def date2ts(date):
    return int(time.mktime(date.timetuple())) * 1000
```