

When squirrels get together for a party, they like to have cigars. A squirrel party is successful when the number of cigars is between 40 and 60, inclusive. Unless it is the weekend, in which case there is no upper bound on the number of cigars. Return True if the party with the given values is successful, or False otherwise.

```
In [ ]: def squirrel_party(cigars, is_weekend):
        if is_weekend:
            return cigars >= 40
        else:
            return 40 <= cigars <= 60

print(squirrel_party(30, False))
print(squirrel_party(50, False))
print(squirrel_party(70, True))
print(squirrel_party(30, True))
print(squirrel_party(45, False))
print(squirrel_party(45, True))
```

You are driving a little too fast, and a police officer stops you. Write code to compute the result, encoded as an int value: 0=no ticket, 1=small ticket, 2=big ticket. If speed is 60 or less, the result is 0. If speed is between 61 and 80 inclusive, the result is 1. If speed is 81 or more, the result is 2. Unless it is your birthday -- on that day, your speed can be 5 higher in all cases.

```
In [ ]: def calculate_ticket(speed, is_birthday):
        if is_birthday:
            speed -= 5

        if speed <= 60:
            return 0
        elif speed <= 80:
            return 1
        else:
            return 2

print(calculate_ticket(60, False))
print(calculate_ticket(65, False))
print(calculate_ticket(85, False))
print(calculate_ticket(60, True))
print(calculate_ticket(65, True))
print(calculate_ticket(85, True))
```

The number 6 is a truly great number. Given two int values, a and b, return True if either one is 6. Or if their sum or difference is 6. Note: the function abs(num) computes the absolute value of a number.

```
In [ ]: def is_great_number(a, b):  
        if a == 6 or b == 6:  
            return True  
        if a + b == 6:  
            return True  
        if abs(a - b) == 6:  
            return True  
        return False  
  
print(is_great_number(6, 4))  
print(is_great_number(4, 6))  
print(is_great_number(2, 4))  
print(is_great_number(8, 2))  
print(is_great_number(7, 1))  
print(is_great_number(12, 6))  
print(is_great_number(1, 5))
```

You and your date are trying to get a table at a restaurant. The parameter "you" is the stylishness of your clothes, in the range 0..10, and "date" is the stylishness of your date's clothes. The result getting the table is encoded as an int value with 0=no, 1=maybe, 2=yes. If either of you is very stylish, 8 or more, then the result is 2 (yes). With the exception that if either of you has style of 2 or less, then the result is 0 (no). Otherwise the result is 1 (maybe).

```
In [ ]: def get_table(you, date):  
        if you <= 2 or date <= 2:  
            return 0  
        if you >= 8 or date >= 8:  
            return 2  
        return 1  
  
print(get_table(5, 10))  
print(get_table(5, 2))  
print(get_table(5, 5))  
print(get_table(8, 3))  
print(get_table(2, 8))  
print(get_table(7, 7))  
print(get_table(3, 9))  
print(get_table(0, 5))
```

Given 2 ints, a and b, return their sum. However, sums in the range 10..19 inclusive, are forbidden, so in that case just return 20.

```
In [ ]: def custom_sum(a, b):
        total = a + b
        if 10 <= total <= 19:
            return 20
        return total

print(custom_sum(3, 4))
print(custom_sum(9, 4))
print(custom_sum(10, 10))
print(custom_sum(5, 5))
print(custom_sum(6, 3))
print(custom_sum(15, 2))
```

The squirrels in Palo Alto spend most of the day playing. In particular, they play if the temperature is between 60 and 90 (inclusive). Unless it is summer, then the upper limit is 100 instead of 90. Given an int temperature and a boolean is_summer, return True if the squirrels play and False otherwise.

```
In [ ]: def squirrels_play(temperature, is_summer):
        upper_limit = 100 if is_summer else 90
        return 60 <= temperature <= upper_limit

print(squirrels_play(70, False))
print(squirrels_play(95, False))
print(squirrels_play(95, True))
print(squirrels_play(101, True))
print(squirrels_play(60, False))
print(squirrels_play(90, False))
print(squirrels_play(100, True))
print(squirrels_play(59, False))
```

Given a day of the week encoded as 0=Sun, 1=Mon, 2=Tue, ...6=Sat, and a boolean indicating if we are on vacation, return a string of the form "7:00" indicating when the alarm clock should ring. Weekdays, the alarm should be "7:00" and on the weekend it should be "10:00". Unless we are on vacation -- then on weekdays it should be "10:00" and weekends it should be "off"

```
In [ ]: def alarm_clock(day, on_vacation):  
    if on_vacation:  
        if day in [0, 6]:  
            return "off"  
        else:  
            return "10:00"  
    else:  
        if day in [0, 6]:  
            return "10:00"  
        else:  
            return "7:00"  
  
print(alarm_clock(0, False))  
print(alarm_clock(1, False))  
print(alarm_clock(5, False))  
print(alarm_clock(6, False))  
print(alarm_clock(0, True))  
print(alarm_clock(2, True))  
print(alarm_clock(6, True))  
print(alarm_clock(4, True))
```

Given a non-negative number "num", return True if num is within 2 of a multiple of 10. Note: (a % b) is the remainder of dividing a by b, so (7 % 5) is 2. See also: Introduction to Mod

```
In [ ]: def near_multiple_of_10(num):  
    remainder = num % 10  
    return remainder in [0, 1, 2, 8, 9]  
  
print(near_multiple_of_10(12))  
print(near_multiple_of_10(17))  
print(near_multiple_of_10(20))  
print(near_multiple_of_10(22))  
print(near_multiple_of_10(29))  
print(near_multiple_of_10(30))  
print(near_multiple_of_10(38))  
print(near_multiple_of_10(41))
```

```
In [ ]:
```