```
import sys

# create a function that will take a question and an answer
# from a true/false question and display it to the user
# the function will take the user's response and return
# whether they answered it correctly as a boolean value
def display_truefalse(question, answer):
    user_answer = input(question + "₩nIs this statement true or false? (T/F): ")
    if user_answer == answer:
        return True
    return False

# create a function that will take a question, an answer, and choices
# and display a multiple choice question to the user
# the function will take the user's response and return whether they
# answered it correctly as a boolean value
def display_multiplechoice(question, answer, choices):
    if len(choices) == 4:
        print(question)
        i = 1
        for select in choices:
            print("%i) %s" % (i, select))
            i = i + 1

        user_answer = input("Enter your selection: ")
        if user_answer == answer:
            return True
        return False
    else:
        raise ValueError

# create a function that will take a line of quiz data as a parameter
# and call one of the previous display functions based on the 'type'
# of question (defined by the first value in the line of quiz data)
# the function should return the boolean values returned from the
# display functions
def display_question(line):
    data = line.rstrip('₩n').split(',')
    if data[0] == 'TF':
        ret = display_truefalse(data[1], data[2])
    else:
        ret = display_multiplechoice(data[1], data[2], *[data[3:7]])

    if ret:
        print("Correct!")
```

```
        else:
            print(f"Incorrect. The answer is {data[2]}")
        return ret

    # make a function that creates a line of the quiz file
    # for a true/false question. the function will ask the
    # user to enter the question and what the correct answer is
    def create_truefalse():
        question = input("Enter the question: ")
        answer = input("Enter the answer (T or F): ")
        return f"TF,{question},{answer.upper()}"

    # make a function that creates a line of the quiz file
    # for a multiplel choice question. the function will ask
    # the user to enter the question, the different choices
    # to pick from (ENTER to end) and the correct answer
    def create_multiplechoice():
        i = 1
        choices = list()
        question = input("Enter the question: ")
        while True:
            possible_answer = input("Enter a possible ansswer (Enter to end): ")
            if possible_answer == '':
                if i > 4:
                    break
                else:
                    continue
            choices.append(possible_answer)
            i = i + 1

        # show a multiple choice question you have made
        print(question)
        i = 1
        for select in choices:
            print("%i) %s" % (i, select))
            i = i + 1

        str_choices = ",".join(choices)
        answer = input("Which one is the correct answer: ")
        return f"MC,{question},{answer},{str_choices}"

    # make a function that will ask the user which type of question
    # they want to create (MC or TF), then call the appropriate
    # create function above. the function will return the line
    # of data that is returned from the other create functions
    def create_question():
```

```
    input_type = input("What type of question do you want create (MC, TF or Enter to end)? ")
    type = input_type.upper()
    if type == 'TF':
        question = create_truefalse()
    elif type == 'MC':
        question = create_multiplechoice()
    elif type == '':
        question = type
    else:
        question = 'invalid'
    return question

# create a function that will open the file for reading
# the function will read in each line of the file
# and call "display_question" with the contents of a line
# the function should return the total number of correct answers
def run_quiz(filename):
    num = 0
    correct = 0
    f = open(filename)
    for line in f:
        num = num + 1
        ret = display_question(line)
        if ret:
            correct = correct + 1
        print("--")
    f.close()

    rate = correct / num * 100
    print(f"You have {correct}/{num} ({rate}%) correct.")

# make a function that creates a new, empty file using the filename passed in as
# a parameter. the function will then call 'create_question' to add new questions
# to the file until the user has finished
# (ensure that the file is closed properly)
def create_quiz(filename):
    # open quiz.tt
    f = open(filename, 'w')

    # create questions
    while True:
        question = create_question()
        if question == '':
            print("Quiz created!")
            break
        elif question == 'invalid':
```

```
            continue
        else:
            f.write(question)
            f.write('₩n')

    # close file
    f.close()

# start the program by creating a "main" function that reads in the two command
# line parameters the program should exit if there are invalid parameters.
# the first parameter will be either 'create' or 'run', the second parameter will
# be a filename. the function will call either "create_quiz" or "run_quiz"
# depending on the parameter passed
def print_help_exit():
    print()
    print('Usage: quiz.py <create|run> <filename>')
    sys.exit(-1)

def main():
    if len(sys.argv) > 2:
        if sys.argv[1] == "create":
            create_quiz(sys.argv[2])
        elif sys.argv[1] == "run":
            run_quiz(sys.argv[2])
        else:
            print_help_exit()
    else:
        print_help_exit()

if __name__ == "__main__":
    main()
```