===== Intro =====

The Knimbus Boston office is located in Boston’s financial district. MBTA (Massachusetts Bay Transportation Authority) is a public transportation system serving the Massachusetts Bay area. Many engineers in the Boston office use the Park Street Station to get to/from the office.

As engineers in the office will tell you, the MBTA system is infamously off schedule. Luckily, the MBTA public API (https://www.mbta.com/developers/v3-api) has a real-time information feed containing information about real-time estimated departures for specific stations. Your goal is to write a small program that utilizes the MBTA API that will quickly tell :

- The current time
- The next 10 trains leaving Park Street Station
- Where they are going (destination town/city)
- Minutes until scheduled departure

Rules/constraints:

- Print the trains grouped by which line they are using, and in the order that they are leaving the station
- Limit the number of trains printed to the 10 trains next leaving the station. Do not print trains that have already left the station (negative departure time).
- Print the number of minutes until the scheduled departure time.
- Print the destination town/city of each train
- You can use any standard or third party libraries available in coderpad.

Example output:

```
‘‘‘
<Current Time>
----Red Line----
Ashmont/Braintree: Departing in 3 minutes
Alewife: Departing in 5 minutes
Ashmont/Braintree: Departing in 9 minutes

----Green Line C----
North Station: Departing in 4 minutes
Cleveland Circle: Departing in 6 minutes

----Green Line E----
North Station: Departing in 4 minutes
Heath Street: Departing in 5 minutes

----Green Line D----
Riverside: Departing in 5 minutes
Government Center: Departing in 7 minutes

----Green Line B----
Park Street: Departing in 6 minutes

‘‘‘
```

Your output does not need to match this, this is just an example.
If you have better ideas of how to display the data, please do!
Just include the required information.

Consider this to be production level code and
should include, but not limited to, exception handling, error
checking, and diagnostic/logging information. You should consider
execution performance, memory consumption, and general efficiency
in your design.

Finally, please help us by keeping this question and your
answer secret so that every candidate has a fair chance in
future interviews.

===== Steps =====

1. Choose the language you want to code in from the menu labeled "Plain Text" in the top right corner of the screen. You will see a "Run" button appear on the top left -- clicking this will send your code to a Linux server and compile / run it. Output will appear on the right side of the screen.

   For information about what libraries are available for your chosen language, see:

   https://coderpad.io/languages

2. Pull up the documentation for the API you'll be using:

   https://api-v3.mbta.com/docs/swagger/index.html#/Prediction/ApiWeb_PredictionController_index

   https://www.mbta.com/developers/v3-api

3. Since the above API doesn't have the name of the routes, or the destination name we are interested in, you'll want to include Route Relationships in the query. To help you save time figuring out the stop to filter over, use "place-pktrm". This is the MBTA's id for the Park Street Station.

   Suggested parameters for the API call:

   include=route -- Used to get additional route information

   filter[stop]=place-pktrm -- Used to filter the data to Park Street Station

   sort=departure_time -- Used to sort by 'departure_time' instead of the default 'arrival_time'

   Tip: You can refer following endpoint for getting required data (feel free to modify it as per your need and optimization):
   https://api-v3.mbta.com/predictions/?filter[stop]=place-pktrm&sort=departure_time&include=route

4. Implement the functionality described above, using data fetched dynamically from the MBTA API endpoint here:

   https://api-v3.mbta.com/predictions

5. Writing tests for your code is optional. If you do write tests for your code, add them to the main() method of your program so that we can easily run them.

====== FAQs =====

Q: What timezone does the MBTA API use?
A: They use UTC-4. For your solution, it would make the most sense to compare arrival times in the same time zone.

Q: What does a direction_id of 0 or 1 mean?
A: This is used as a key to identify which direction (0 or 1) the train is headed in. You can use this key with the direction_destinations in the included route data to find the destination of the train.

Q: How do I know if my solution is correct?
A: Make sure you've read the assignment carefully and you're convinced your program does what you think it should in the common case. If your program does what the assignment dictates, you will get full credit. We do not use an auto-grader, so we do not have any values for you to check correctness against.

Q: What is Knimbus looking for in a solution?
A: After submitting your code, we'll have a pair of engineers evaluate it and determine next steps in the interview process. We are looking for correct, easy-to-read, robust code. Specifically, ensure your code is idiomatic and laid out logically. Ensure it is correct. Ensure it handles all edge cases and error cases elegantly.

Q: If I need a clarification, who should I ask?
A: Send all questions to the email address that sent you this document, and an engineer at Knimbus will get back to you ASAP (we're pretty quick during normal business hours).

Q: How long should this question take me?
A: Approximately 1 hour, but it could take more or less depending on your experience with web APIs and the language you choose.

Q: When is this due?
A: We will begin grading your answer 24 hours after it is sent to you, so that is the deadline.

```
*/
```