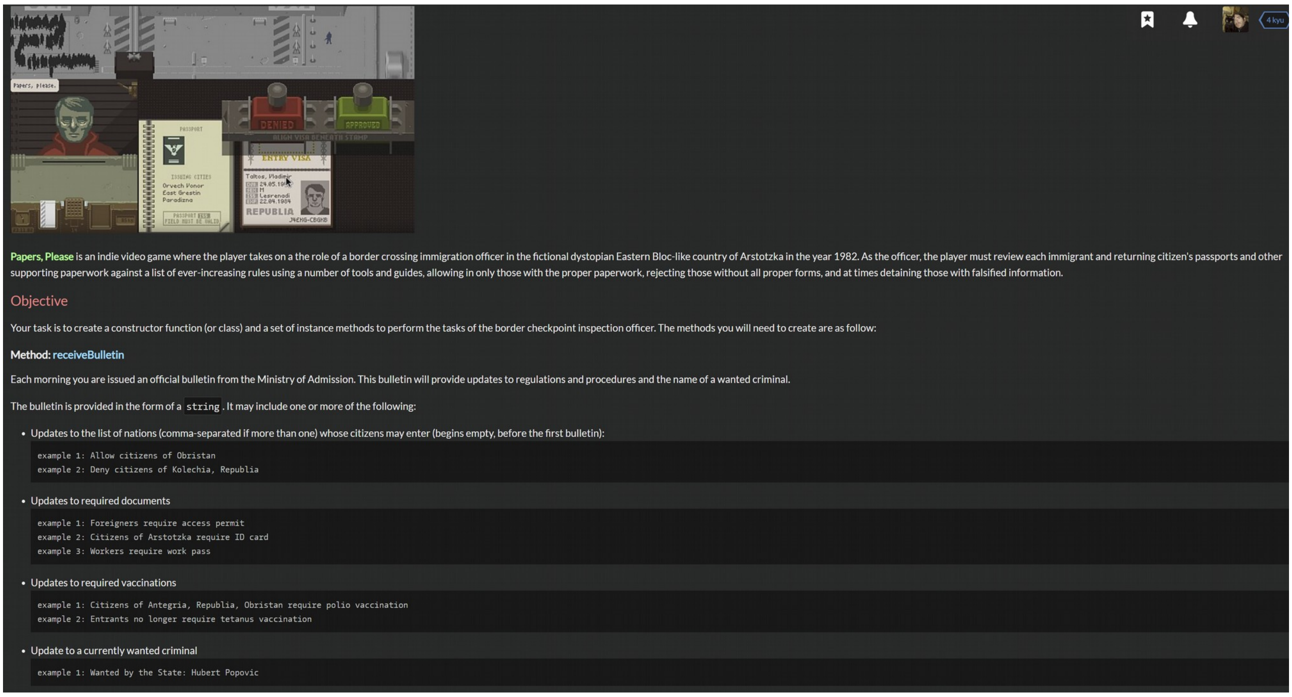


Papers, Please is an indie video game where the player takes on a the role of a border crossing immigration officer in the fictional dystopian Eastern Bloc-like country of Arstotzka in the year 1982. As the officer, the player must review each immigrant and returning citizen's passports and other supporting paperwork against a list of ever-increasing rules using a number of tools and guides, allowing in only those with the proper paperwork, rejecting those without all proper forms, and at times detaining those with falsified information.

Objective

Your task is to create a constructor function (or class) and a set of instance methods to perform the tasks of the border checkpoint inspection officer. The methods you will need to create are as follow:

Method: receiveBulletin

Each morning you are issued an official bulletin from the Ministry of Admission. This bulletin will provide updates to regulations and procedures and the name of a wanted criminal.

The bulletin is provided in the form of a `string`. It may include one or more of the following:

- Updates to the list of nations (comma-separated if more than one) whose citizens may enter (begins empty, before the first bulletin):

  ```
  example 1: Allow citizens of Obristan
  example 2: Deny citizens of Kolechia, Republia
  ```

- Updates to required documents

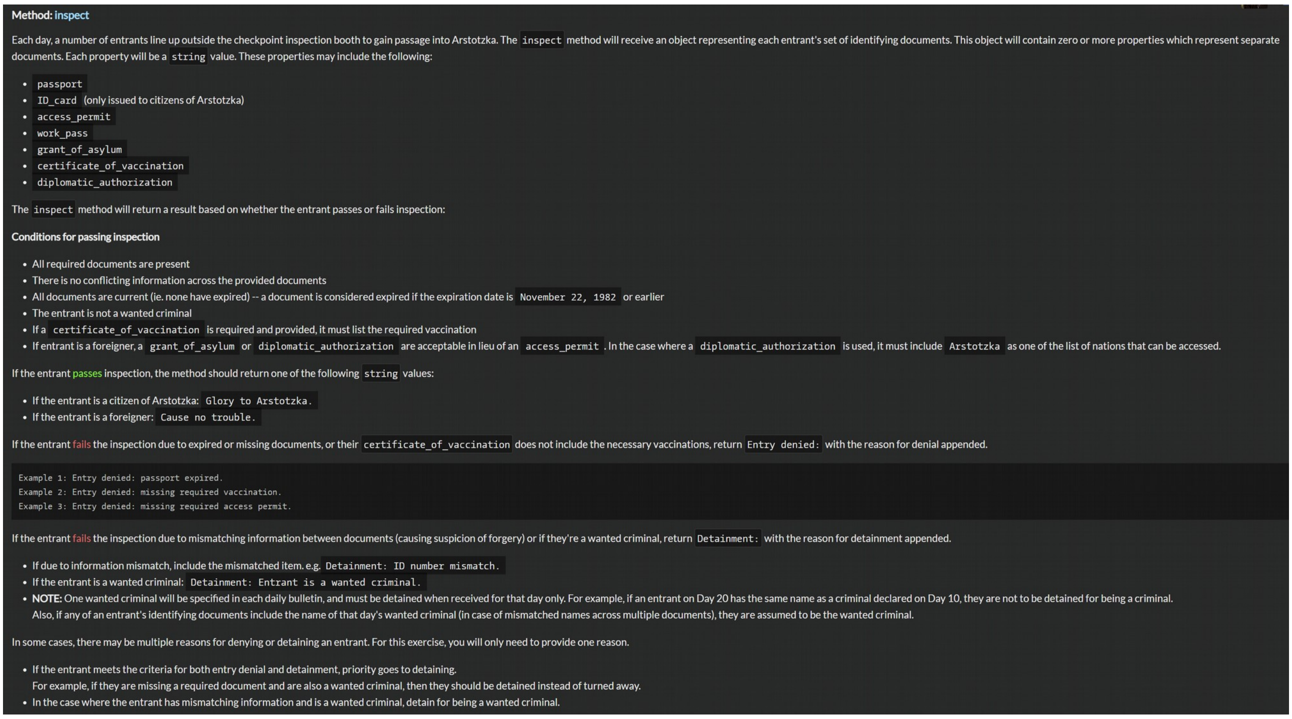
  ```
  example 1: Foreigners require access permit
  example 2: Citizens of Arstotzka require ID card
  example 3: Workers require work pass
  ```

- Updates to required vaccinations

  ```
  example 1: Citizens of Antegria, Republia, Obristan require polio vaccination
  example 2: Entrants no longer require tetanus vaccination
  ```

- Update to a currently wanted criminal

  ```
  example 1: Wanted by the State: Hubert Popovic
  ```



Method: inspect

Each day, a number of entrants line up outside the checkpoint inspection booth to gain passage into Arstotzka. The `inspect` method will receive an object representing each entrant's set of identifying documents. This object will contain zero or more properties which represent separate documents. Each property will be a `string` value. These properties may include the following:

- `passport`
- `ID_card` (only issued to citizens of Arstotzka)
- `access_permit`
- `work_pass`
- `grant_of_asylum`
- `certificate_of_vaccination`
- `diplomatic_authorization`

The `inspect` method will return a result based on whether the entrant passes or fails inspection:

Conditions for passing inspection

- All required documents are present
- There is no conflicting information across the provided documents
- All documents are current (ie. none have expired) -- a document is considered expired if the expiration date is `November 22, 1982` or earlier
- The entrant is not a wanted criminal
- If a `certificate_of_vaccination` is required and provided, it must list the required vaccination
- If entrant is a foreigner, a `grant_of_asylum` or `diplomatic_authorization` are acceptable in lieu of an `access_permit`. In the case where a `diplomatic_authorization` is used, it must include `Arstotzka` as one of the list of nations that can be accessed.

If the entrant passes inspection, the method should return one of the following `string` values:

- If the entrant is a citizen of Arstotzka: `Glory to Arstotzka.`
- If the entrant is a foreigner: `Cause no trouble.`

If the entrant fails the inspection due to expired or missing documents, or their `certificate_of_vaccination` does not include the necessary vaccinations, return `Entry denied:` with the reason for denial appended.

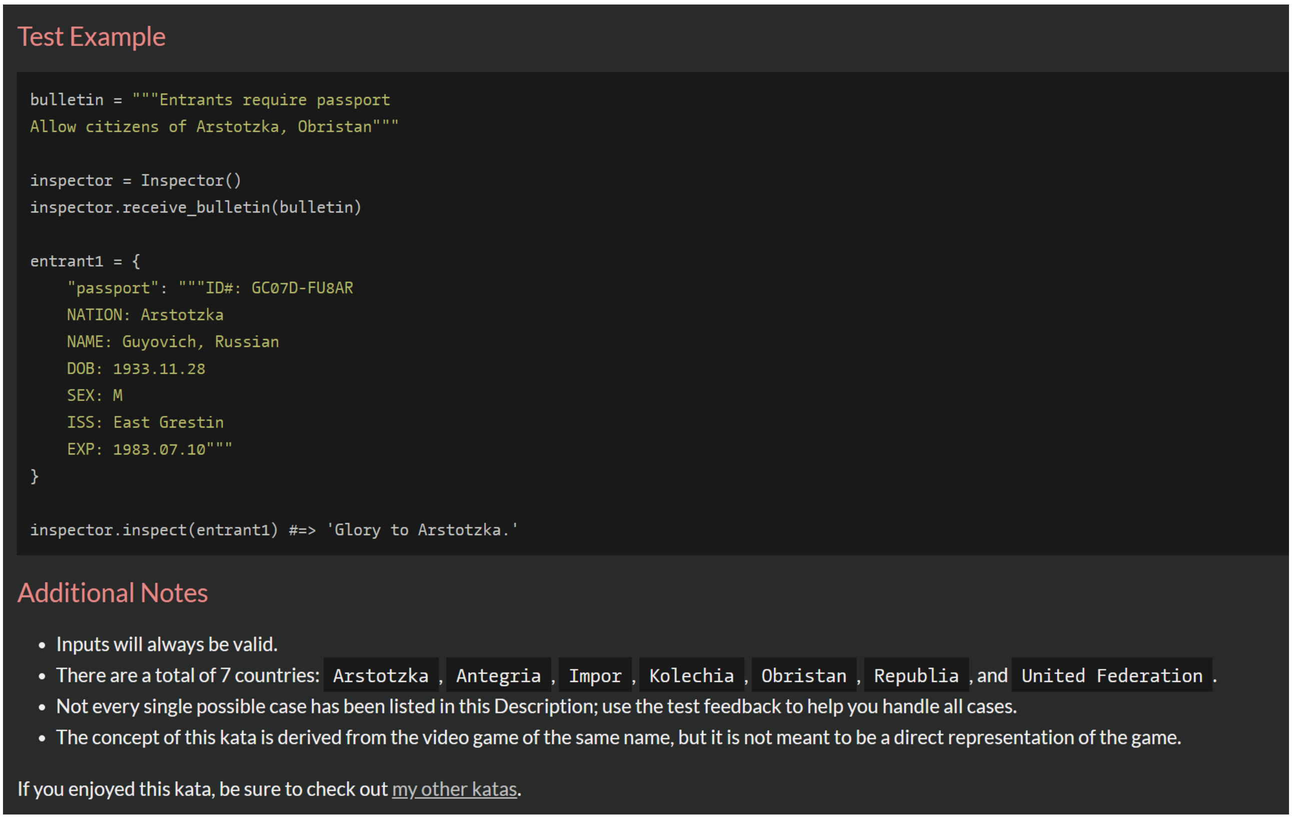
```
Example 1: Entry denied: passport expired.
Example 2: Entry denied: missing required vaccination.
Example 3: Entry denied: missing required access permit.
```

If the entrant fails the inspection due to mismatching information between documents (causing suspicion of forgery) or if they're a wanted criminal, return `Detainment:` with the reason for detainment appended.

- If due to information mismatch, include the mismatched item. e.g. `Detainment: ID number mismatch.`
- If the entrant is a wanted criminal: `Detainment: Entrant is a wanted criminal.`
- **NOTE:** One wanted criminal will be specified in each daily bulletin, and must be detained when received for that day only. For example, if an entrant on Day 20 has the same name as a criminal declared on Day 10, they are not to be detained for being a criminal.
  Also, if any of an entrant's identifying documents include the name of that day's wanted criminal (in case of mismatched names across multiple documents), they are assumed to be the wanted criminal.

In some cases, there may be multiple reasons for denying or detaining an entrant. For this exercise, you will only need to provide one reason.

- If the entrant meets the criteria for both entry denial and detainment, priority goes to detaining.
  For example, if they are missing a required document and are also a wanted criminal, then they should be detained instead of turned away.
- In the case where the entrant has mismatching information and is a wanted criminal, detain for being a wanted criminal.



Test Example

```
bulletin = """Entrants require passport
Allow citizens of Arstotzka, Obristan"""

inspector = Inspector()
inspector.receive_bulletin(bulletin)

entrant1 = {
    "passport": """ID#: GC07D-FU8AR
    NATION: Arstotzka
    NAME: Guyovich, Russian
    DOB: 1933.11.28
    SEX: M
    ISS: East Grestin
    EXP: 1983.07.10"""
}

inspector.inspect(entrant1) #=> 'Glory to Arstotzka.'
```

Additional Notes

- Inputs will always be valid.
- There are a total of 7 countries: `Arstotzka`, `Antegria`, `Impor`, `Kolechia`, `Obristan`, `Republia`, and `United Federation`.
- Not every single possible case has been listed in this Description; use the test feedback to help you handle all cases.
- The concept of this kata is derived from the video game of the same name, but it is not meant to be a direct representation of the game.

If you enjoyed this kata, be sure to check out my other katas.