```
"""Module containing all the data structures for parties in cases."""

class Party(object):
    """The base object from which all other party objects are inherited."""
    def __init__(self, party_type, party_name: str):
        self.party_type = party_type # Either Person or Entity


class PlaintiffEntity(Party):
    """A type of party in a case. Generally the State of Ohio in a criminal
    case."""
    def __init__(self, party_type: 'Entity'):
        super().__init__(party_type)
        self.full_name = party_type.entity_name


class Person(object):
    """A base class for all people of any party type."""
    def __init__(self) -> None:
        self.first_name = None
        self.last_name = None
        self.date_of_birth = None


class Defendant(Person):
    """Class for defendants that inherits from Person class."""
    def __init__(self):
        super().__init__()


class JudicialOfficer(Person):
    """
    A subclass of a person that is for all different types of judicial
    officers: Judges, Visiting Judges, Acting Judges, Magistrates.

    In the future may want to add attorney registration number to this class.
    """
    def __init__(self, first_name: str, last_name: str, officer_type: str) -> None:
        super().__init__()
        self.first_name = first_name
        self.last_name = last_name
        self.officer_type = officer_type


class Entity(object):
    """A base class for non-person parties
```

```
47          (i.e. State of Ohio or a business)."""
48
49          def __init__(self, entity_name: str):
50              self.entity_name = entity_name
51
52
53      """
54      test = JudicialOfficer("Marianne", "Hemmeter", "Judge")
55      print(test.officer_type)
56      test_2 = PlaintiffEntity(Entity("State of Ohio"))
57      print(test_2.party_type)
58      print(test_2.full_name)
59      """
```