```
<one Entity class per service in the schema, with informable \pm requestable slots>
class Attraction(Entity):
       find touristy stuff to do around you
       Parameters:
        area: (str) area to search for attractions (e.g. 'east', ...)
        name: (str) name of the attraction
       type: (str) type of the attraction (e.g. 'college', 'boat', ...)
       entrance fee: (str) how much is the entrance fee
       open_hours: (str) hours of the attraction
        address: (str) address of the attraction
        phone: (str) phone number of the attraction
        postcode: (str) postal code of the attraction
   <parameters as fields, default=None>
    . . .
< a class for each of the acts supported in our system>
class Inform(Act):
   """ A service act communicates something about a service Entity"""
   entity: Entity = None
class Request(Act):
   """Ask for specific information or action."""
   values: List[str] = None
if name == ' main ':
   agent = DialogueAgent()
   # Provide the dialogue acts corresponding to the observed system response
   # Example 1
   response = agent.handle turn(
        system response="Ok, where will you be departing from?",
        system acts=[Request(values=['departure'])]
```