EXPERIMENT 1

PROBLEM TITLE

Implement a Simple Reflex Agent and a Model-Based Reflex Agent in a Simulated Environment

CODE

```
class SimpleReflexAgent:
    def __init__(self):
        pass

    def act(self, percept):
        if percept == 'dirt':
            return 'suck'
        else:
            return 'move'

class ModelBasedReflexAgent:
    def __init__(self):
        self.state = 'clean'

    def update_state(self, percept):
        self.state = percept

    def act(self):
        if self.state == 'dirt':
            return 'suck'
        else:
            return 'move'

# Simulation
percepts = ['clean', 'dirt', 'clean', 'dirt']
simple_agent = SimpleReflexAgent()
model_agent = ModelBasedReflexAgent()

print("Simple Reflex Agent Actions:")
for percept in percepts:
    print(f"Percept: {percept}, Action: {simple_agent.act(percept)}")
print("\nModel-Based Reflex Agent Actions:")
for percept in percepts:
    model_agent.update_state(percept)
    print(f"State: {model_agent.state}, Action: {model_agent.act()}")
```

EXPECTED OUTPUT

```
Simple Reflex Agent Actions:
Percept: clean, Action: move
Percept: dirt, Action: suck
Percept: clean, Action: move
Percept: dirt, Action: suck

Model-Based Reflex Agent Actions:
State: clean, Action: move
State: dirt, Action: suck
State: clean, Action: move
State: dirt, Action: suck
```