```
──────────────────── MODULE fowler ────────────────────
Controller of the secret compartment of Mrs. H, who loves secrets! Following the example of M.
Fowler, which can be found at: http://martinfowler.com/

Variables
VARIABLE
  state,   the state for display, only to be compliant with Fowler
  light,   state of the light
  draw,    state of the draw
  panel,   state of the secret panel
  door     state of the entry door

Type invariance
TypeInv ≜
  ∧ state ∈ {"idle", "active", "waitingForDraw",
             "waitingForLight", "unlockedPanel"}
  ∧ light  ∈ {"on", "off"}
  ∧ draw ∈ {"opened", "closed"}
  ∧ door  ∈ {"locked", "unlocked"}
  ∧ panel ∈ {"locked", "unlocked"}
Initial state
Init ≜
  ∧ state  = "idle"
  ∧ light  = "off"
  ∧ draw  = "closed"
  ∧ door  = "unlocked"
  ∧ panel = "locked"
├──────────────────────────────────────────────────────┤
Action definition. Note that the state variable is not used for the determination of the actual
state, but only for display. This shows that this variable is not required in TLA+

Closes the door, to activate
CloseDoor ≜
  ∧ door = "unlocked"
  ∧ door′ = "locked"
  ∧ state′ = "active"
  ∧ UNCHANGED ⟨panel, light, draw⟩
Switch on the light, if the draw is opened, this opens the secret panel
LightOn ≜
  ∧ light = "off"
  ∧ light′ = "on"
  ∧ IF draw = "opened" THEN
        ∧ state′ = "unlockedPanel"
            ∧ panel′ = "unlocked"
            $\land\ door' = \text{“locked”}$
          ELSE
            $\land\ state' = \text{“waitingForDraw”}$
            $\land$ UNCHANGED $\langle panel,\ door \rangle$
  $\land$ UNCHANGED $\langle draw \rangle$
```

Open the draw, if the light is on, this opens the secret panel

```
$OpenDraw\ \triangleq$
  $\land\ draw = \text{“closed”}$
  $\land\ draw' = \text{“opened”}$
  $\land$ IF $light = \text{“on”}$ THEN
        $\land\ state' = \text{“unlockedPanel”}$
            $\land\ panel' = \text{“unlocked”}$
            $\land\ door' = \text{“locked”}$
          ELSE
            $\land\ state' = \text{“waitingForLight”}$
            $\land$ UNCHANGED $\langle panel,\ door \rangle$
  $\land$ UNCHANGED $\langle light \rangle$
```

Closes the secret panel and move the system to initial state

```
$ClosePanel\ \triangleq$
  $\land\ panel = \text{“unlocked”}$
  $\land\ panel' = \text{“locked”}$
  $\land\ light' \ = \text{“off”}$
  $\land\ draw' = \text{“closed”}$
  $\land\ door' = \text{“unlocked”}$
  $\land\ state' = \text{“idle”}$
```

All possible actions

```
$Next\ \triangleq$
  $\lor\ CloseDoor$
  $\lor\ LightOn$
  $\lor\ OpenDraw$
  $\lor\ ClosePanel$
```

Specification of the entire system

$$Spec\ \triangleq\ Init \land \Box[Next]_{\langle panel,\, light,\, draw,\, door,\, state \rangle}$$

Specification never violates the type invariance

THEOREM $Spec \Rightarrow \Box\, TypeInv$

The panel and door are never both unlocked in the same time

```
$Inv\ \triangleq$
  $\lor\ panel = \text{“unlocked”} \Rightarrow door = \text{“locked”}$
  $\lor\ door = \text{“unlocked”} \Rightarrow panel = \text{“locked”}$
```