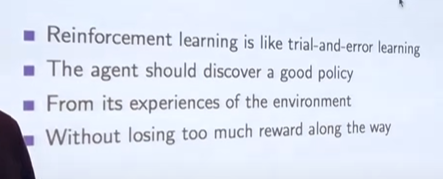
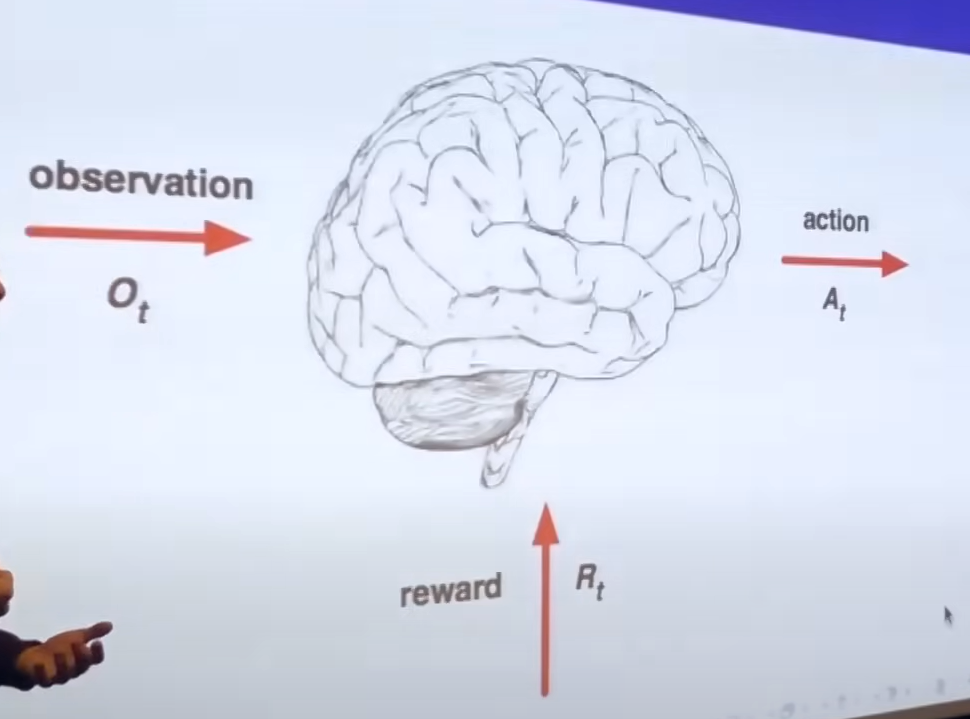
## horizontal line

Reinforcement Learning

16.10.2025

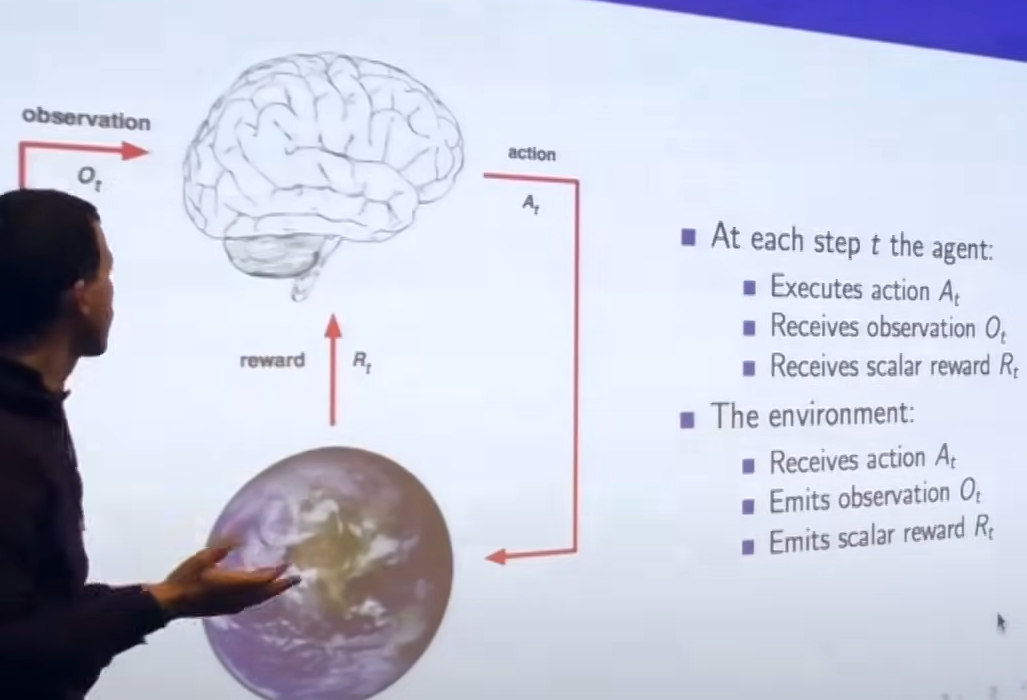
RL typically stands for Reinforcement Learning, a type of machine learning where an agent learns to make decisions by interacting with an environment to maximize some notion of cumulative reward.



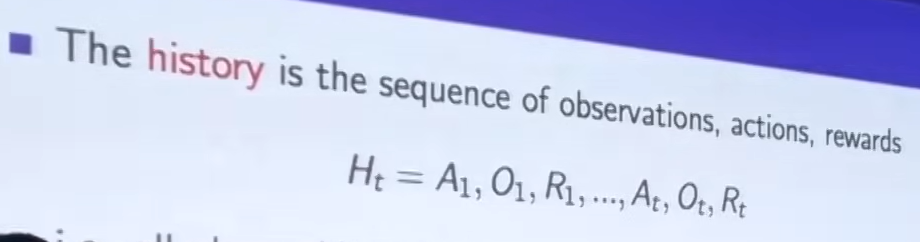


A reward is a scalar feedback signal. Indicates how well the agent is doing at step t. The agent’s job is to maximize the reward.

GOAL - maximisation of expected cumulative reward.

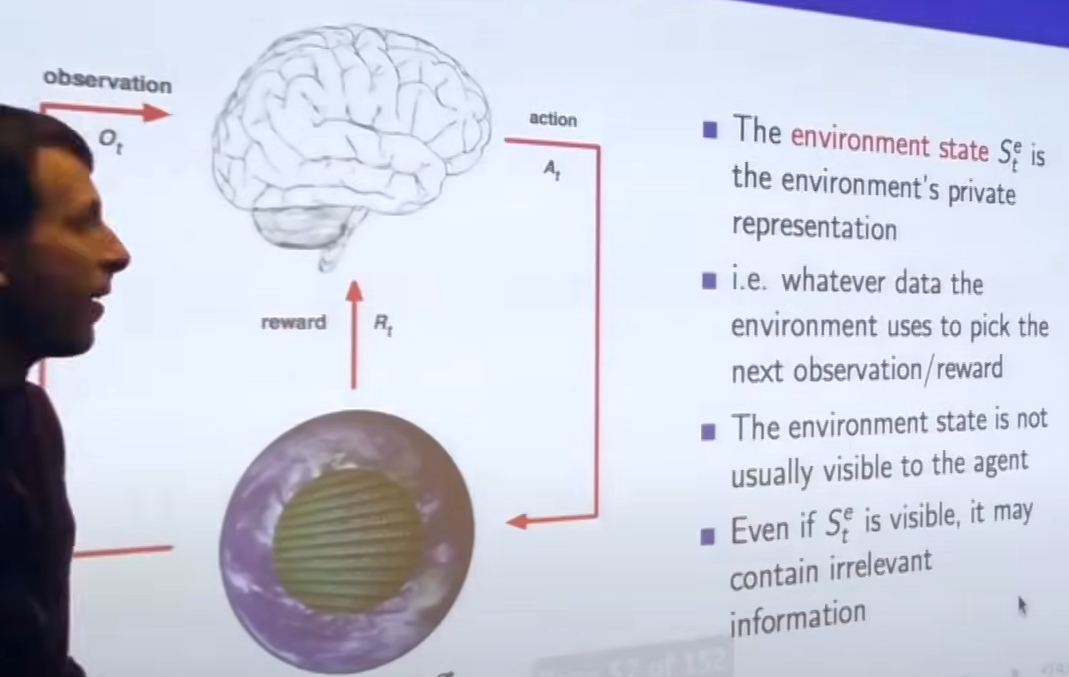


What happens to do next depends on the history :

1. The agent selects action
2. The environment selects observations / rewards 

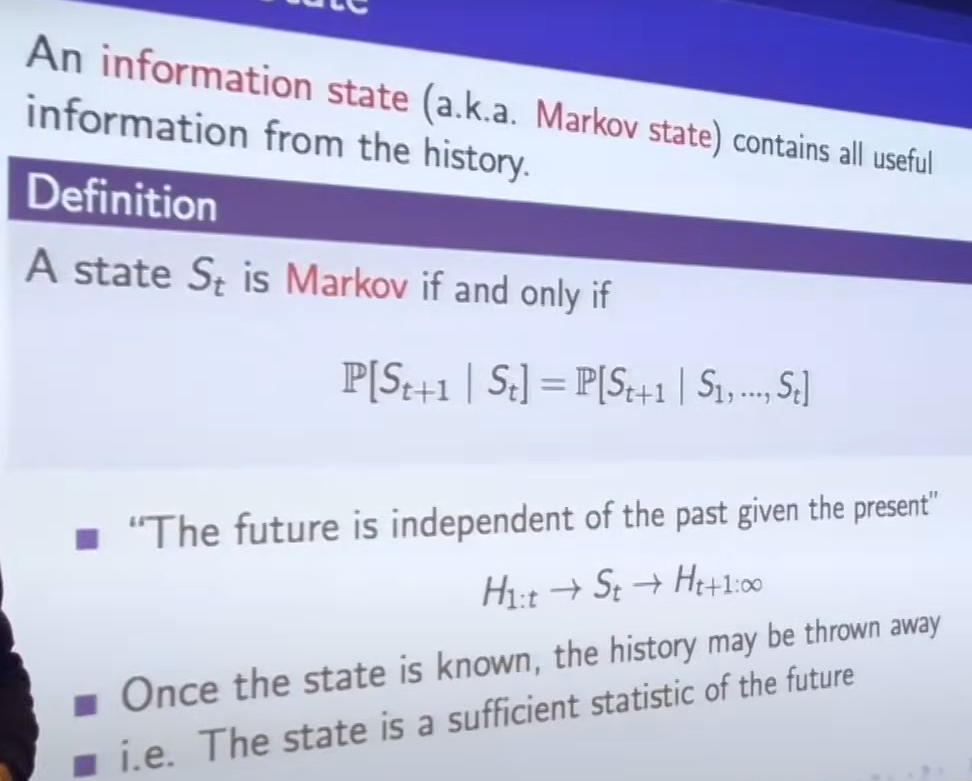
State is the information used to determine what happens next , extracted from history , updated in between to do not carry history for long instead having State for next prediction

Formally , state is a function of history : St = f (Ht)

**Environment’s State**



**Agent’s State**



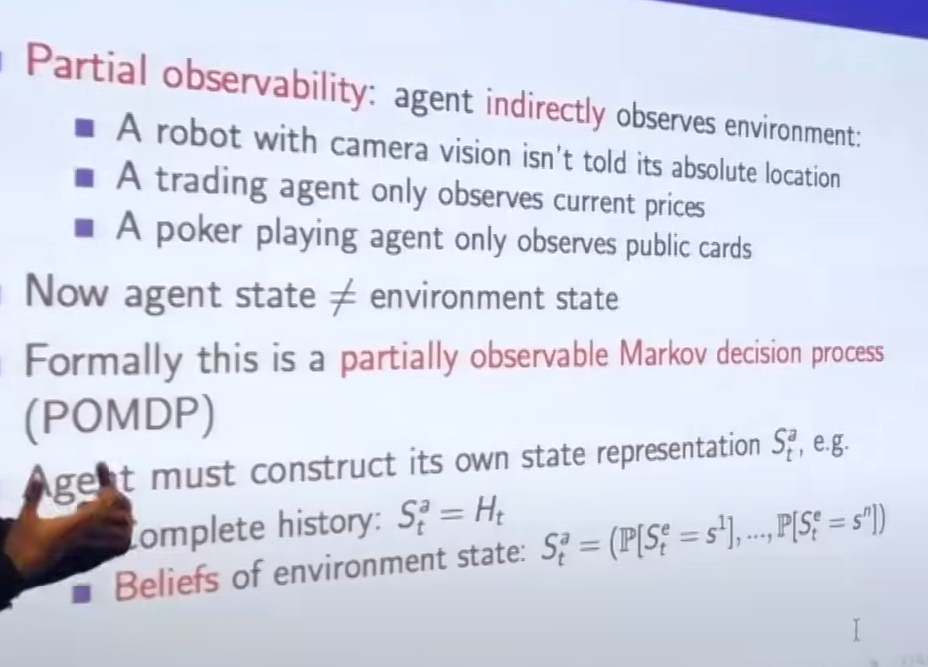
Markov State - contains all useful information from the history

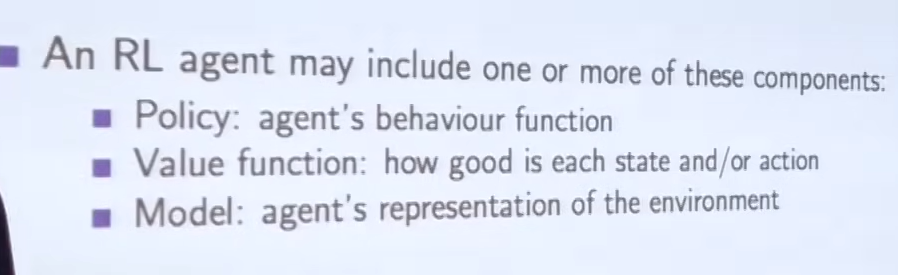
Environment State is S is Markov and the History Ht is Markov.

1. Fully Observability Environments : the best case

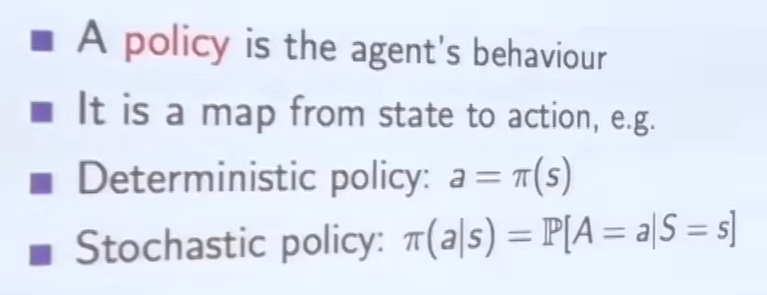


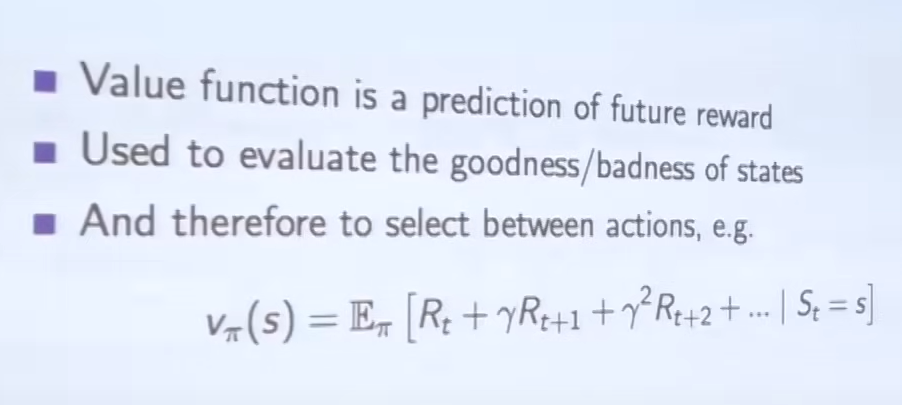
1. Partially Observability Environments :



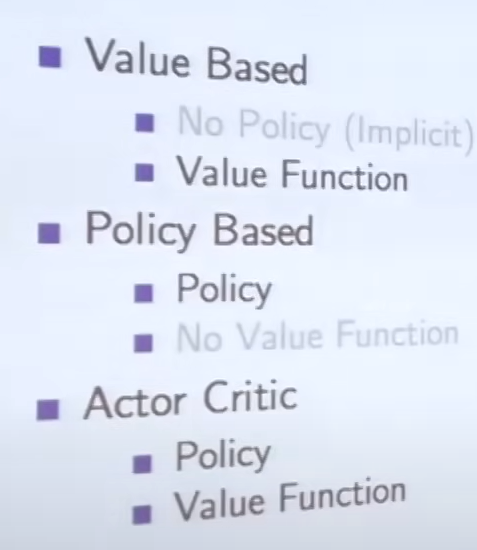
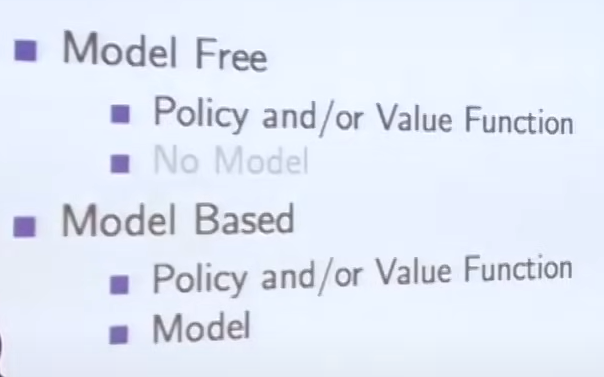


# Policy , Value Function , Model





# Categorizing RL Agents :



# Exploration Vs Exploitation

**In reinforcement learning (RL), *exploration* means trying new actions to discover better outcomes, while *exploitation* means choosing the best-known action to maximize reward. Balancing both is key to learning an optimal policy.**

🔎 **Exploration**

* The agent tries **unfamiliar actions** to learn more about the environment.
* Helps discover **new states, rewards, or strategies**.
* Example: A scheduler tries allocating PRBs to a low-priority UE to see if it improves slice performance.

💡 **Exploitation**

* The agent chooses the action that has **given the highest reward so far**.
* Focuses on **maximizing immediate gains**.
* Example: A scheduler keeps prioritizing Slice A because it consistently yields high throughput.

**In reinforcement learning (RL), *bootstrapping* means updating value estimates using other estimates instead of waiting for full outcomes, while *sampling* means learning from actual experiences collected during interaction with the environment. Together, they shape how agents learn efficiently and adaptively.**

🧠 Bootstrapping in RL

**Bootstrapping** refers to the idea of:

**Using your current predictions to improve future predictions.**

Instead of waiting for the full return (like in Monte Carlo), bootstrapping updates value estimates using **partial information** — typically the value of the next state.

🎲 Sampling in RL

**Sampling** means:

**Learning from actual experiences** — states, actions, rewards — collected during interaction with the environment.

Instead of computing expectations over all possible transitions (like in Dynamic Programming), RL agents **sample one path** and learn from it.

🔁 Bootstrapping vs Sampling

Most modern RL algorithms use **both**:

* **TD learning** uses bootstrapping and sampling
* **Monte Carlo** uses sampling only
* **Dynamic Programming** uses bootstrapping only (with full model)