Union

↳ the same for struct → different Handle Memory

↳ create

typedef union Player

{
  uint8_t Name[20];
  uint8_t No;
} Real Players;

→ Create Union Variable — New data type

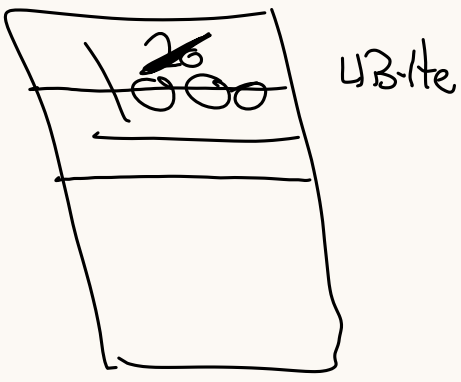
→ ① Real Players P1;
② Union Player P2;

→ ① Real Players P1 = {"Ben", 6};

↳ Handle Memory

Union Player
{
  char x;
  int ex;
};

share the same memory.



union Player P1;

P1.X = 20

P1.(ex) = 1000

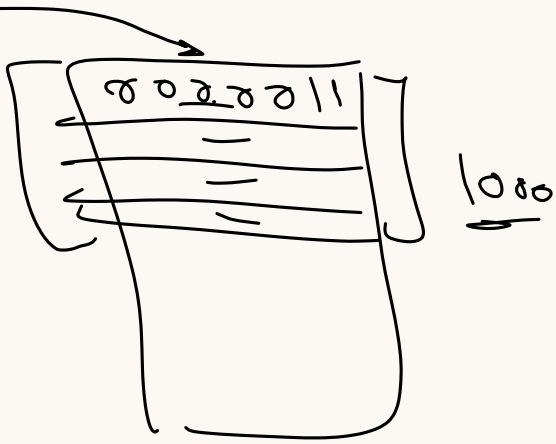
→ Garbage

→ 1000

$P_1.x$ ⟶ garage

$P_1.t$ ⟶ 1000

$P_1.x = 3$



$P_1.t =$

ATM

- RFID
- SIM

Union Atm

- RFID
- SIM

Enum

↳ userdefine type

↳ set name for Constant

int arr[5]

size

100

50

10

```
#define lowSpeed
#define HighSpeed
#define NormalSpeed
```

→ enum DCMotorSpeed

{ lowSpeed = 10,

HighSpeed = 100, NormalSpeed = 50,

typedef enum
{

} Configration;