Enum [illegible]

[illegible] Enum [illegible]

```
enum colors {RED, GREEN, BLUE};


printf("%d\n", RED); // 0
printf("%d\n", GREEN);  // 1
printf("%d\n", BLUE);  // 2
```

[illegible] `enum` [illegible] `colors` [illegible] `RED`[illegible]`GREEN`[illegible]`BLUE` [illegible] `0`[illegible]`1`[illegible]`2` [illegible] `RED` [illegible] `0` [illegible]

[illegible] Enum [illegible]

[illegible] Enum [illegible]

```
enum colors color;
```

[illegible] `color` [illegible] `enum colors` [illegible] `RED`[illegible]`GREEN`[illegible]`BLUE` [illegible]

```
color = BLUE;
printf("%i\n", color); // 2
```

[illegible] `color` [illegible] `BLUE` [illegible] `BLUE` [illegible] `2`[illegible]

typedef [illegible] Enum [illegible]

```
typedef enum {
  SHEEP,
  WHEAT,
  WOOD,
  BRICK,
  ORE
} RESOURCE;
```

```
RESOURCE r;
```

[illegible] `RESOURCE` [illegible] Enum [illegible]

[illegible] Enum [illegible]

```
enum {
  SHEEP,
  WHEAT,
  WOOD,
  BRICK,
  ORE
} r = BRICK, s = WOOD;
```

[illegible] `r` [illegible] `3` [illegible] `s` [illegible] `2` [illegible]

[illegible] Enum [illegible] Enum [illegible]

```
enum { ONE, TWO };


printf("%d %d", ONE, TWO);  // 0 1
```

[illegible] `enum` [illegible] `ONE` [illegible] `TWO` [illegible] Enum [illegible]

[illegible]

```
enum { ONE, TWO, };
```

[illegible] Enum [illegible] C [illegible] 0 [illegible] C [illegible] ENUM [illegible] Enum [illegible]

```
enum { ONE = 1, TWO = 2 };


printf("%d %d", ONE, TWO);  // 1 2
```

Enum [illegible]

```
enum { X = 2, Y = 18, Z = -2 };
```

Enum [illegible]

```
enum { X = 2, Y = 2, Z = 2 };
```

[illegible]
[illegible]

```
enum {
  A,    // 0
  B,    // 1
  C = 4,  // 4
  D,    // 5
  E,    // 6
  F = 3,   // 3
  G,    // 4
  H     // 5
};
```

Enum
[illegible]
[illegible] `int` [illegible] Enum [illegible]

[illegible] #1
[illegible] [illegible] [illegible] 2021-09-04 22:34:34 UTC
[illegible] [illegible] [illegible] 2022-06-30 13:08:38 UTC