



## EXPLORER

- ✓ app
- > public
- > segments
- > services
- ✓ src
  - ✓ domain
    - keepHelloCount.ts
    - sayHello.ts
  - > webui
- index.html
- package.json

## CODE



## TERMINAL





## Code Editor

### EXPLORER

- ✓ app
- > public
- > segments
- > services
- ✓ src
  - ✓ domain
    - keepHelloCount.ts
    - sayHello.ts
- > webui
- index.html
- package.json

### CODE

```
<>
```

### TERMINAL

```
>
```

## EXPLORER

- ✓  app
- >  public
- >  segments
- >  services
- ✓  src
- ✓  domain
  -  keepHelloCount.ts
  -  sayHello.ts
- >  webui
-  index.html
-  package.json

## CODE



## TERMINAL

&gt;

## EXPLORER

- ✓  app
- >  public
- >  segments
- >  services
- ✓  src
- ✓  domain
  -  keepHelloCount.ts
  -  sayHello.ts
- >  webui
-  index.html
-  package.json

## CODE



## TERMINAL

&gt;

# Code Editor

## EXPLORER

- ✓  app
- >  public
- >  segments
- >  services
- ✓  src
- ✓  domain
  -  keepHelloCount.ts
  -  sayHello.ts
- >  webui
-  index.html
-  package.json

## CODE

## TERMINAL

>



## Code Editor

### EXPLORER

- ✓ app
- > public
- > segments
- > services
- ✓ src
  - ✓ domain
    -  keepHelloCount.ts
    -  sayHello.ts
  - > webui
-  index.html
-  package.json

CODE → src/domain/keepHelloCount.ts

### TERMINAL



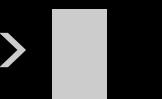
## Code Editor

CODE → src/domain/keepHelloCount.ts

### EXPLORER

- /  app
- >  public
- >  segments
- >  services
- ✓  src
- ✓  domain
  -  keepHelloCount.ts
  -  sayHello.ts
- >  webui
-  index.html
-  package.json

### TERMINAL



R

pp

public

segments

services

src

| domain

s | keepHelloCount.ts

s | sayHello.ts

| webui

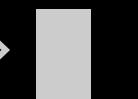
index.html

package.json

## Code Editor

CODE → src/domain/keepHelloCount.ts

TERMINAL



## Code Editor

CODE → src/domain/keepHelloCount.ts

nts

s

n

HelloCount.ts



Hello.ts

TERMINAL



ml

e.json

## Code Editor

CODE → src/domain/keepHelloCount.ts

unt.ts

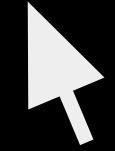


TERMINAL



## Code Editor

CODE → src/domain/keepHelloCount.ts



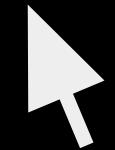
## TERMINAL



## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
  
export async function keepHelloCount(name: string): Promise<number>  
{  
  helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
  return helloCounts[name];  
}
```



## TERMINAL



## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
  
export async function keepHelloCount(name: string): Promise<number>  
{  
  helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
  return helloCounts[name];  
}
```



## TERMINAL



## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
    return helloCounts[name];  
}
```



## TERMINAL

> █

## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
    return helloCounts[name];  
}
```

## TERMINAL

> █

## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
    return helloCounts[name];  
}
```

## TERMINAL

> █

## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
    return helloCounts[name];  
}
```

## TERMINAL

> █

## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
    return helloCounts[name];  
}
```

## TERMINAL

> █

## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
    return helloCounts[name];  
}
```

## TERMINAL

> █

## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
    return helloCounts[name];  
}
```

## TERMINAL

> █

## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
    return helloCounts[name];  
}
```

## TERMINAL

> █

## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
    return helloCounts[name];  
}
```

## TERMINAL

> █

## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
    return helloCounts[name];  
}
```

## TERMINAL

> █

## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
    return helloCounts[name];  
}
```

## TERMINAL

> █

## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
    return helloCounts[name];  
}
```

## TERMINAL

> █

## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
      
    return helloCounts[name];  
}
```

## TERMINAL



## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
    return helloCounts[name];  
}
```

## TERMINAL

> █

## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
    return helloCounts[name];  
}
```

## TERMINAL

> █

## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
    return helloCounts[name];  
}
```

## TERMINAL

> █

## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
    return helloCounts[name];  
}
```

## TERMINAL

> █

## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
    return helloCounts[name];  
}
```

## TERMINAL

> █

## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
    return helloCounts[name];  
}
```

## TERMINAL

> █

## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
    return helloCounts[name];  
}
```

## TERMINAL

> █

## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
    return helloCounts[name];  
}
```

## TERMINAL

> █

## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
    return helloCounts[name];  
}
```

## TERMINAL

> █

## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
    return helloCounts[name];  
}
```

## TERMINAL

> █

## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
    return helloCounts[name];  
}
```

## TERMINAL

> █

## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
    return helloCounts[name];  
}
```

## TERMINAL

> █

## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};
export async function keepHelloCount(name: string): Promise<number>
{
  helloCounts[name] = (helloCounts[name] ?? 0) + 1;

  return helloCounts[name];
}
```

## TERMINAL

> █

## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
    return helloCounts[name];  
}
```

## TERMINAL

> █

## Code Editor

CODE → src/domain/keepHelloCount.ts

```
const helloCounts: Record<string, number> = {};  
export async function keepHelloCount(name: string): Promise<number>  
{  
    helloCounts[name] = (helloCounts[name] ?? 0) + 1;  
  
    return helloCounts[name];  
}
```

## TERMINAL

> █