# 스크린샷 관련 서비스

특정 모니터 또는 특정 프로세스 창의 스크린샷을 획득 또는 저장하기 위한 기능들을 제공합니다.

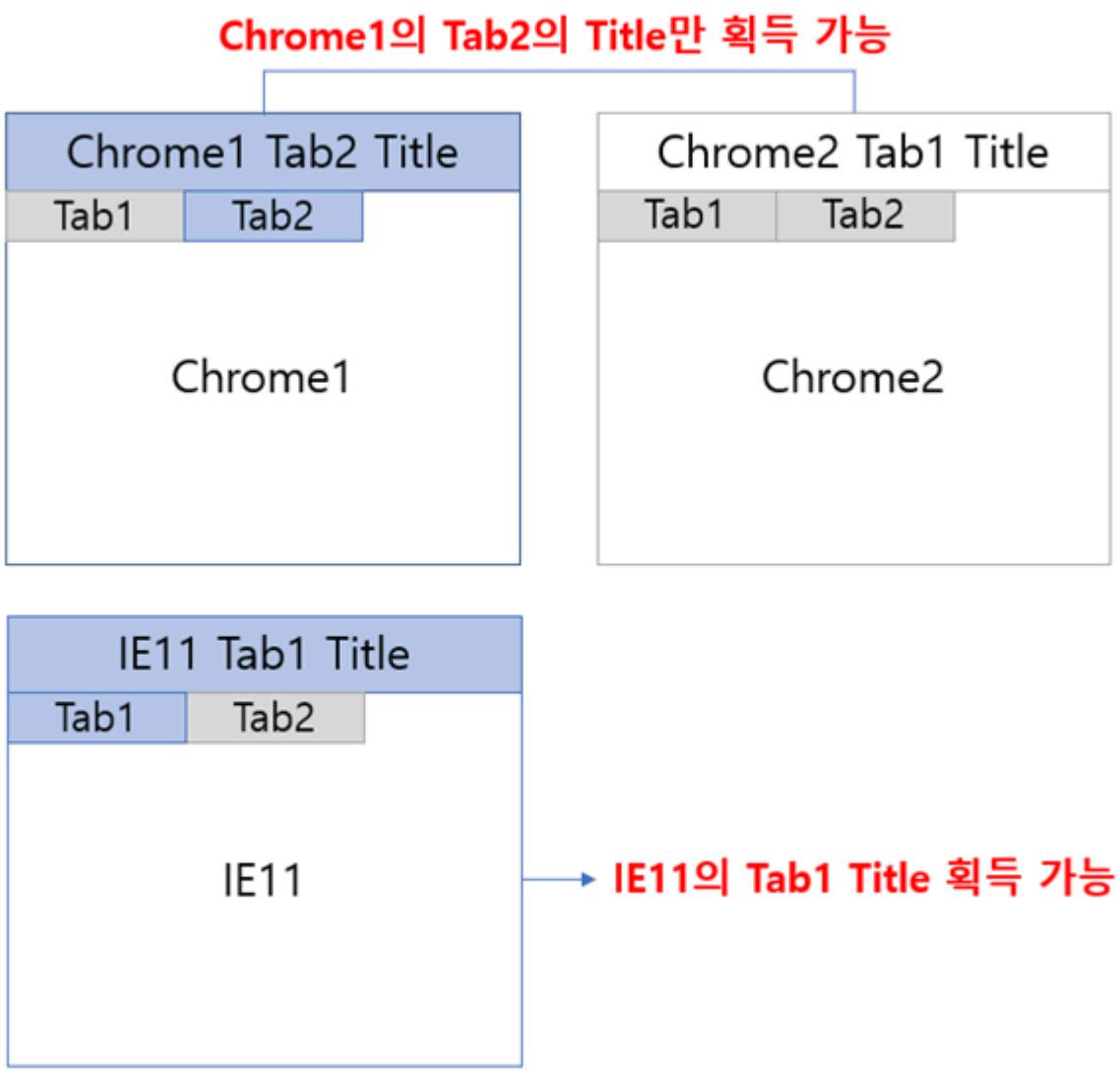
본 서비스에서 eXDevice+가 인식하는 프로세스의 창 영역은 다음과 같습니다.




**<eXDevice+가 인식하는 프로세스 창 영역>**

또한 웹 브라우저의 경우 Windows 정책 상 현재 포커스 된 창의 포커스 된 탭의 타이틀만 식별할 수 있습니다. 즉, 특정 브라우저에 포커스가 가 있지 않은 경우 해당 브라우저의 마지막으로 오픈한 창의 오픈된 탭의 타이틀을 검색할 수 있습니다.

Copyright ⓒ TOMATO SYSTEM CO., LTD. All rights reserved.




**<eXDevice+ 브라우저 타이틀 인식 예시>**

# 서비스 리스트

스크린샷 관련 서비스 리스트는 다음과 같습니다.

| No. | 서비스명 | API명 | 비고 |
|---|---|---|---|
| 1 | 모니터 스크린샷 획득 | WinInfo.GetScreenShot | |
| 2 | 모니터 스크린샷 저장 | WinInfo.SaveScreenShot | |
| 3 | 타이틀 기반 프로세스 창 스크린샷 획득 | WinInfo.GetWindowScreenShot | |

Copyright ⓒ TOMATO SYSTEM CO., LTD. All rights reserved.

| No. | 서비스명 | API명 | 비고 |
|---|---|---|---|
| 4 | 타이틀 기반 프로세스 창 스크린샷 저장 | WinInfo.SaveWindowScreenShot | |

# 서비스 상태 코드 리스트

본 서비스에서 공통으로 사용되는 서비스 상태코드 리스트에 대해 정의합니다.

이 상태 코드는 응답 JSON 메시지 중 *return* 또는 *returnValue*의 값에 해당합니다.

| 리턴코드 | 리턴 코드 내용 | 대상 서비스 | 비고 |
|---|---|---|---|
| -1 | 정의되지 않은 예외 | 공통 | Log 파일에서 내용 확인 필요 |
| 0 | 정상 | 공통 | - |
| 2 | PC와 연결된 모니터 리스트 획득 실패 | GetScreenShot, SaveScreenShot | |
| 3 | 요청 모니터 정보 획득 실패 | GetScreenShot, SaveScreenShot | |
| 6 | 대상 타이틀을 가진 프로세스가 없음 | GetWindowScreenShot, SaveWindowScreenShot, GetProcessScreenShot, SaveProcessScreenShot | |
| 10 | 창 최소화 모드에서 스크린샷 획득 불가 | GetWindowScreenShot, SaveWindowScreenShot, GetProcessScreenShot, SaveProcessScreenShot | |
| 11 | 스크린샷 획득 실패 | GetWindowScreenShot, SaveWindowScreenShot, GetProcessScreenShot, SaveProcessScreenShot | |
| 12 | 파일 저장 실패 | SaveScreenShot, SaveWindowScreenShot, SaveProcessScreenShot | |

# 1. 모니터 스크린샷 획득

**API명**

- WinInfo.GetScreenShot

**정의**

Copyright ⓒ TOMATO SYSTEM CO., LTD. All rights reserved.

- 지정한 모니터의 화면을 캡처하고, 캡처한 이미지를 Base64 String으로 획득합니다.

> ✅ 모니터와 관련된 정보는 *"모니터 관련 서비스"* 문서를 참고하시길 바랍니다.

**호출예시**

```
{
  "service":"WinInfo.GetScreenShot",
  "requestKey":"Random Request Key",
  "param":{
    "screenIndex": 4
  }
}
```

- screenIndex
  - 화면을 캡처할 모니터 인덱스.
  - 시스템이 인식하는 모니터 명칭에서 숫자 값. (예: DISPLAY4 → 4)

**응답예시**

```
{
  "service":"WinInfo.GetScreenShot",
  "requestKey":"호출시 사용한 Request Key",
  "statusCode":"0000",
  "return":{
    "returnValue":0,
    "screenShotBase64":"/9j/4AAQSkZJRgABAQEAYABgAAD/2wBDAAMCAgMCAgMDAwMEA..."
  }
}
```

- returnValue
  - 서비스 상태 코드.

- screenShotBase64
  - Base64 String 데이터로 인코딩 된 모니터 스크린샷 데이터.

Copyright ⓒ TOMATO SYSTEM CO., LTD. All rights reserved.

# 2. 모니터 스크린샷 저장

**API명**

- WinInfo.SaveScreenShot

**정의**

- 지정한 모니터의 화면을 캡처하고, 캡처한 이미지를 로컬 디스크에 저장합니다.

> 모니터와 관련된 정보는 *"모니터 관련 서비스"* 문서를 참고하시길 바랍니다.

호출예시

```
{
  "service": "WinInfo.SaveScreenShot",
  "requestKey": "Random Request Key",
  "param": {
    "screenIndex": 4,
    "filePath": "D:\\test.jpg"
  }
}
```

- screenIndex
  - 화면을 캡처할 모니터 인덱스.
  - 시스템이 인식하는 모니터 명칭에서 숫자 값. (예: DISPLAY4 → 4)
- filePath
  - 스크린샷을 저장할 로컬 파일 경로.

**응답예시**

```
{
  "service":"WinInfo.SaveScreenShot",
  "requestKey":"호출시 사용한 Request Key",
  "statusCode":"0000",
```

Copyright ⓒ TOMATO SYSTEM CO., LTD. All rights reserved.

```
  "return":0
}
```

- return
  - 서비스 상태 코드.

# 3. 타이틀 기반 프로세스 창 스크린샷 획득

**API명**

- WinInfo.GetWindowScreenShot

**정의**

- 타이틀을 가지는 프로세스의 창(Window)을 찾고, 해당 창을 캡쳐한 이미지를 Base64 String으로 획득합니다.

> ⚠ 프로세스의 창이 최소화 모드인 경우 해당 창의 스크린샷을 획득할 수 없습니다.

**호출예시**

```
{
  "service":"WinInfo.GetWindowScreenShot",
  "requestKey":"Random Request Key",
  "param": {
    "processTitle":"TOMATOSYSTEM"
  }
}
```

- processTitle
  - 캡쳐할 프로세스 창의 타이틀.

**응답예시**

Copyright ⓒ TOMATO SYSTEM CO., LTD. All rights reserved.

```
{
  "service":"WinInfo.GetWindowScreenShot",
  "requestKey":"호출시 사용한 Request Key",
  "statusCode":"0000",
  "return":{
    "returnValue":0,
    "screenShotBase64":"/9j/4AAQSkZJRgABAQEAYABgAAD/2wBDAA..."
  }
}
```

- returnValue
  - 서비스 상태 코드.

- screenShotBase64
  - Base64 String 데이터로 인코딩 된 프로세스 창 스크린샷 데이터.

# 4. 타이틀 기반 프로세스 창 스크린샷 저장

**API명**

- WinInfo.SaveWindowScreenShot

**정의**

- 타이틀을 가지는 프로세스의 창(Window)을 찾고, 해당 창을 캡처한 이미지를 로컬 디스크에 저장 합니다.

⚠ 프로세스의 창이 최소화 모드인 경우 해당 창의 스크린샷을 획득할 수 없습니다.

**호출예시**

```
{
  "service":"WinInfo.SaveWindowScreenShot",
  "requestKey":"Random Request Key...",
  "param":{
```

Copyright ⓒ TOMATO SYSTEM CO., LTD. All rights reserved.

```
    "processTitle":"TOMATOSYSTEM",
    "filePath":"C:\\test.jpg"
  }
}
```

- processTitle
  - 캡쳐할 프로세스 창의 타이틀.
- filePath
  - 스크린샷을 저장할 로컬 파일 경로.

**응답예시**

```
{
  "service":"WinInfo.SaveWindowScreenShot",
  "requestKey":"호출시 사용한 Request Key",
  "statusCode":"0000",
  "return":0
}
```

- return
  - 서비스 상태 코드.

# 5. PID 기반 프로세스 창 스크린샷 획득

**API명**

- WinInfo.GetProcessScreenShot

**정의**

- 특정 프로세스 ID(PID)를 가지는 프로세스 창(Window)을 찾고, 해당 창을 캡쳐한 이미지를 Base64 String으로 획득합니다.

Copyright ⓒ TOMATO SYSTEM CO., LTD. All rights reserved.

⚠ 프로세스의 창이 최소화 모드인 경우 해당 창의 스크린샷을 획득할 수 없습니다.

**호출예시**

```
{
  "service":"WinInfo.GetProcessScreenShot",
  "requestKey":"Random Request Key",
  "param":{
    "processId":1234
  }
}
```

- processId
  - 검색할 프로세스 ID.

**응답예시**

```
{
  "service":"WinInfo.GetProcessScreenShot",
  "requestKey":"호출시 사용한 Request Key",
  "statusCode":"0000",
  "return":{
    "returnValue":0,
    "screenShotBase64":"/9j/4AAQSkZJRgABAQEAYABgAAD/2w..."
  }
}
```

- returnValue
  - 서비스 상태 코드.

- screenShotBase64
  - Base64 String 데이터로 인코딩 된 프로세스 창 스크린샷 데이터.

# 6. PID 기반 프로세스 창 스크린샷 저장

Copyright ⓒ TOMATO SYSTEM CO., LTD. All rights reserved.

### API명

- WinInfo.SaveWindowScreenShot

### 정의

- 특정 프로세스 ID(PID)를 가지는 프로세스 창(Window)을 찾고, 해당 창을 캡쳐한 이미지를 로컬 디스크에 저장합니다.

### 호출예시

```
{
  "service":"WinInfo.SaveProcessScreenShot",
  "requestKey":"Random Request Key",
  "param":{
    "processId":1234,
    "filePath":"C:\\test.jpg"
  }
}
```

- processId
  - 검색할 프로세스 ID.
- filePath
  - 스크린샷을 저장할 로컬 경로.

### 응답예시

```
{
  "service":"WinInfo.SaveProcessScreenShot",
  "requestKey":"호출시 사용한 Request Key",
  "statusCode":"0000",
  "return":0
}
```

- return
  - 서비스 상태 코드.

Copyright ⓒ TOMATO SYSTEM CO., LTD. All rights reserved.

Copyright ⓒ TOMATO SYSTEM CO., LTD. All rights reserved.