

SJC-P 静的インポート

- ・静的インポートは、import static とする。
- ・クラス名.静的要素 とすることで、配下の静的要素をインポートできる。(1)
- ・クラス名.* とすることで、配下の静的要素を一括インポートできる。(2)
- ・曖昧な場合、コンパイルエラー (3)
- ・競合する場合でも、明示的にインポートしたものがあれば、そちらが利用される。(3)
- ・同一メソッド名が存在しても、呼び出すメソッドは引数で判定される。(4)

```
package staticimport;

import static java.lang.System.out;           // System クラスの静的 out オブジェクトをインポート (1)
import static java.lang.Math.*;              // オブジェクト配下の静的要素をすべてインポート (2)
import static java.lang.Integer.MAX_VALUE;    // 定数 (static final) をインポート (3)
import static java.lang.Long.*;
import static java.lang.Double.*;

public class StaticImportTest1 {
    public static void main(String[] args) {
        // System.out とする必要がなくなる
        out.println(Math.PI);                  // Math.PI が利用できる
        out.println(toHexString(MAX_VALUE));    // Integer.MAX_VALUE が、Long.*、Double.* より強い (3)
    }
    // out.println(toHexString(MIN_VALUE));      // NG コンパイルエラー MIN_VALUE が long か double か
    曖昧 (3)
    out.println(toHexString(Long.MIN_VALUE));    // どの toHexString() が利用されるかは、引数で判定 (4)
}

    for (int i=0x0; i<=0xFF; i++) {
        out.format("%1$02X ", i);
        if (((i+1) % 8) == 0) out.println("");
    }
}
```