

JAVA BEGINNER

EXCEPTION HANDLING



Abstract of New Technology

Website <http://www.antkh.com>

Tel 010 / 016 66 66 53

Prepared By Tum Sakal

Tel 087 36 31 30

Mail tumsakal.ts@gmail.com

Facebook [sak kal](#)

EXCEPTION

- Exception ជាករណីមួយដែលតែងតែកើតឡើង នៅពេលកម្មវិធី ជួបប្រទះបញ្ហា Error ពេលដំនើការ
- នៅពេលមាន Exception កើតឡើង កម្មវិធីនៅគាំង ឬឈប់ដំនើការ
- ដូចនេះអ្នកត្រូវតែសរសេរកូដចាប់ និងដោះស្រាយ Exception នោះ
- **try-catch statement** ត្រូវប្រើប្រាស់ដើម្បីចាប់ និងដោះស្រាយ Exception

TYPE OF EXCEPTION

- Java បែងចែក Exception ជាពីរប្រភេទគឺ:
 - Unchecked Exception
 - និង Checked Exception

UNCHECKED EXCEPTION

- Unchecked Exception ជាប្រភេទ **Runtime Exception** ដែលមិនបានត្រួតពិនិត្យ ថាវានឹងអាចកើតឡើងនៅក្នុងករណីណាមួយច្បាស់លាស់នៅពេលដែលកម្មវិធីកំពុងដំនើការ
- Unchecked Exception មិនទាមទារអោយមានការប្រើប្រាស់ try-catch
- Unchecked Exception គឺជាប្រភេទ Exception Class ដែល extends **java.lang.RuntimeException** មានដូចជា
 - `ArrayIndexOutOfBoundsException`
 - `NegativeArraySizeException` និង `ArithmeticException` ជាដើម

CHECKED EXCEPTION

- Checked Exception ជាប្រភេទ **Exception** ដែលបានត្រួតពិនិត្យថាវានឹងអាចកើតឡើងនៅក្នុងករណីណាមួយច្បាស់លាស់ នៅពេលដែលកម្មវិធីកំពុងដំនើរការ
- Checked Exception ចាំបាច់ត្រូវតែប្រើប្រាស់ **try-catch**
- Checked Exception ជា Exception Class ដែល extends `java.lang.Exception` មានដូចជា
 - SQLException, IOException និង ParseException ជាដើម

TRY-CATCH

```
try{  
    error expected statements..  
} catch(ExceptionA){  
    resolve statements..  
} catch(ExceptionB){  
    resolve statements..  
} catch(ExceptionC | ExceptionD | ...){  
    resolve statements..  
}
```

THROWS EXCEPTION

- ករណីដែលអ្នកមិនគោលបំណងក្នុងការចាប់ និងដោះស្រាយ Exception រឺអ្នកមិនមានលទ្ធភាពក្នុងការសរសេរកូដដោះស្រាយ ទៅលើ Exception នោះអ្នកអាចបន្តបោះ រឺ **throws** Exception

```
method() throws CheckedExceptionA, CheckedExceptionB, ...{  
}
```