
Subject: [Solved]CoreTutorial package problem
Posted by [bozero](#) on Thu, 30 Jan 2020 02:48:47 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi, I have tried the CoreTutorial package and run the programme, but it stopped at tutorial2.cpp line 16:

```
DO(Stream);
```

Would you please look into it. Thanks.

My case is win10 + up-mingw-13664 / 13916.

Subject: Re: CoreTutorial package problem
Posted by [mirek](#) on Fri, 31 Jan 2020 16:23:48 GMT
[View Forum Message](#) <> [Reply to Message](#)

First of all, the whole package is only meant to generate the tutorial document.

Anyway, what do you mean by "stopped"? I have just tried it and it seems to work fine...

Mirek

Subject: Re: CoreTutorial package problem
Posted by [bozero](#) on Sat, 01 Feb 2020 11:19:06 GMT
[View Forum Message](#) <> [Reply to Message](#)

TheIDE pop out a message box as below when the programme is run step by step to CoreTutorial\Stream.cpp line 72.

Message box:

```
===
```

Failed to obtain information about threads. Make sure your application posses debug info. The debugger and debugge proceses will be stoped! [OK]

```
===
```

Part of CoreTutorial\Stream.cpp

```
===
```

```
/// In `LoadThrowing` mode, `Stream` throws `LoadingError` exception when there is problem with  
/// input `Stream`:
```

```
in.LoadThrowing();  
try {  
    in.GetAll(999999); <--- line 72  
}
```

```
catch(LoadingError) {  
    LOG("Loading error");  
}  
===
```

I comment out the above code, rebuild and run the programme.
It run further to CoreTutorial\Serialize.cpp line 42 and pop out the same message box.

Part of CoreTutorial\Serialize.cpp

```
===  
/// When serialization fails to load the data (e.g. because of wrong structure or not  
/// enough data in the stream), `Stream::LoadError` is invoked, which can trigger the  
/// exception if the stream is `LoadThrowing`:  
  
ss2.Seek(0);  
ss2.LoadThrowing();  
try {  
    ss2 % x2 % h2 % x2; <-- line 42  
}  
catch(LoadingError) {  
    LOG("Deserialization has failed");  
}  
  
===
```

Thank you for you attention.

Subject: Re: CoreTutorial package problem
Posted by [mirek](#) on Sat, 01 Feb 2020 13:17:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

bozero wrote on Sat, 01 February 2020 12:19TheIDE pop out a message box as below when the programme is run step by step to CoreTutorial\Stream.cpp line 72.

Message box:

```
===  
Failed to obtain information about threads. Make sure your application posses debug info. The  
debugger and debugge proceses will be stoped! [OK]  
===
```

Part of CoreTutorial\Stream.cpp

```
===  
/// In `LoadThrowing` mode, `Stream` throws `LoadingError` exception when there is problem with  
/// input `Stream`:  
  
in.LoadThrowing();  
try {
```

```
in.GetAll(999999); <--- line 72
}
catch(LoadingError) {
    LOG("Loading error");
}
===
```

I comment out the above code, rebuild and run the programme.
It run further to CoreTutorial\Serialize.cpp line 42 and pop out the same message box.

Part of CoreTutorial\Serialize.cpp

```
===
/// When serialization fails to load the data (e.g. because of wrong structure or not
/// enough data in the stream), `Stream::LoadError` is invoked, which can trigger the
/// exception if the stream is `LoadThrowing`:

ss2.Seek(0);
ss2.LoadThrowing();
try {
    ss2 % x2 % h2 % x2; <-- line 42
}
catch(LoadingError) {
    LOG("Deserialization has failed");
}

===
```

Thank you for you attention.

I see. That is not a problem of CoreTutorial, but (probably) a bug in mingw toolchain - debugger fails to start.

You should be able to run it with "Execute Ctrl+F5".

Mirek

Subject: Re: CoreTutorial package problem
Posted by [Xemuth](#) on Sat, 01 Feb 2020 13:28:41 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello, I tried it with MinGWx64 / MinGW (this one don't open properly the GUI, I guess it's intended) / MSVS19 / MSVSx64
and it work fine.

Win 10 - Version 13954 of TheIDE

Subject: Re: CoreTutorial package problem
Posted by [bozero](#) on Sun, 02 Feb 2020 10:32:29 GMT
[View Forum Message](#) <> [Reply to Message](#)

I had tried two of previous stable release of upp bundle with mingw tool-chain and find the following:

CoreTutorial:

upp 2019.1 upp-mingw-13068.7z

Fail, can run but not work, no tutorial file generated(both debug/release build tried)

upp 2018.1 upp-mingw-11873.7z

OK, run properly, editor is popup with tutorial file open(both debug/release build tried)

As the problem come from try-throw-catch block, I try a similar case CoWorkExceptions and find:

CoWorkExceptions: (LOG amended as RLOG for inspection at release build)

upp 2019.1 upp-mingw-13068.7z

release build + execute : seem OK, console => exitcode: 3221225477, log file => Caught: Worker exception

release build + debug : Fail, popup warning and stop

debug build + execute : Fail, console => terminate called after throwing an instance of 'char const*', exitcode: 3

debug build + debug : Fail, popup warning and stop

upp 2018.1 upp-mingw-11873.7z

release build + execute : OK, console => Caught: Worker exception, exitcode: 0

release build + debug : OK, Log file => Caught: Worker exception

debug build + execute : OK, console => exitcode: 0, log file => Caught: Worker exception

debug build + debug : OK, Log file => Caught: Worker exception

Thank you for your attention.

Subject: Re: CoreTutorial package problem
Posted by [bozero](#) on Wed, 19 Feb 2020 02:54:11 GMT
[View Forum Message](#) <> [Reply to Message](#)

The problem seem solved at upp-mingw-14027.
Thank you for all for your effort.
