**What is the difference between multi-threading and concurrency?**

Concurrency means happening at (about) the same time. As opposed to happening in parallel, truly.  
  
Your computer runs many applications at the same time (concurrently), if there was only one single core CPU. Each application is called a process in the computer world. For obvious reasons - application/process has a very specific boundary - conceptually.  
  
Ability to run multiple processes (applications), which we take for granted is called multi-processing and is implemented even on one single core CPU. The operating system is what takes care of this.  
  
*What multi-threading allows for is concurrency within each application/process.*  
  
An example would be within a word processing program, while one thread is displaying the entered text, another thread could be continually checking for spellings and another for grammar, etc.  
  
~~  
  
When programmers write code, a programmer views the code (sequenced set of instructions to computer) as a single *thread* of execution, weaving through the code.  
  
Do A  
Do B  
Do C  
Do D  
  
What multi-threaded paradigm allows for is the ability for a programmer to write code like this...  
  
Do A and Do B  
Do C  
Do D  
  
Here task A and B happens concurrently or could even happen in parallel. So there are multiple *threads* of execution with a single application.

Ref: <https://www.quora.com/What-is-the-difference-between-multi-threading-and-concurrency>