**Master Interview**

**Stories**

**Company**

**Position**

**Jive Questions**

**Why Jive**

1. People
   1. Ross
   2. Career Fair
   3. Interview Experience
2. Product
   1. Like mine
      1. Wishlist
      2. Shareshare
   2. Jive-N
      1. Social collaboration software
3. Company Vision and Mission and Future possibilities
   1. Pillars
      1. Connect
      2. Communicate
      3. Collaborate
   2. Mission
      1. Empower people and organizations to work better together
   3. Values
      1. Collaboration
      2. Innovation
      3. Results
      4. Integrity
      5. Fun

**Tell me about a time when you had to deal with a difficult situation and it went well. How about one where it went poorly.**

**Well:**

1. Situation: Reorganizing data storage and practices in student life (technical/interpersonal)
2. Issues:
   1. Deleting valuable information
   2. Overcoming staff (15 to 20 year employees)
   3. Finding a solution all staff members would want to use
3. Action:
   1. Got another staff member on board
   2. Laid out over many weeks
   3. Deletion care
4. Result:
   1. Much better
   2. More organized

**Poorly:**

1. Situation:
   1. Designing site decided to do just desktop and mobile later, created a huge mess when redesigning
2. Action:
   1. Read, planned diagramed
   2. Intense rebuild but worth it
3. Result:
   1. Learned a lot always consider long term or any significant code changes
   2. Learned a lot through Google

**Describe a time you made a mistake and how you handled the repercussions.**

1. Situation:
   1. Setting up list serve which triggered reply all
   2. Massive phone calls everyone replying remove
2. Action:
   1. Heard phones ringing, a minute or two later shut down list to stop all emails
   2. Issues apology emails
3. Result:
   1. Praised at staff meeting- had to say I caused it
   2. Be more cautious when implementing new technology (I already am)

**What do you see yourself doing in 5 years? What is your work style? Tell me about a time that you were satisfied with your work.**

1. Satisfied: Shareshare (almost)
2. Career Services Report

**Programming**

1. DNS issues usually result in what?
2. Write a SQL Join

*SELECT Orders.OrderID, Customers.CustomerName, Orders.OrderDate*

*FROM Orders*

*INNER JOIN Customers*

*ON Orders.CustomerID=Customers.CustomerID;*

1. How would you write a SQL SELECT statement?
   1. *SELECT \* FROM Customers;*
2. There were some standard db/sql questions such as differentiate between inner/left outer joins, indexes, etc as well as open ended questions such as tuning strategy/methodology for a slow application query, etc
   1. Inner Join- The INNER JOIN keyword selects all rows from both tables as long as there is a match between the columns in both tables.

*SELECT column\_name(s)*

*FROM table1*

*INNER JOIN table2*

*ON table1.column\_name=table2.column\_name;*



* 1. Left Join- The LEFT JOIN keyword returns all rows from the left table (table1), with the matching rows in the right table (table2). The result is NULL in the right side when there is no match.

*SELECT column\_name(s)*

*FROM table1*

*LEFT JOIN table2*

*ON table1.column\_name=table2.column\_name;*



* 1. Outer Join



* 1. Union- The UNION operator is used to combine the result-set of two or more SELECT statements.Notice that each SELECT statement within the UNION must have the same number of columns. The columns must also have similar data types. Also, the columns in each SELECT statement must be in the same order.

1. How would you troubleshoot a 500 Internal Server error?
   1. Most of the time, "wrong" means an issue with the page or site's programming, but there's certainly a chance the problem is on your end, something we'll investigate below.
   2. Fixing
      1. A Permissions Error. In most cases, a 500 Internal Server Error is due to an incorrect permission on one or more files or folders. In most of those cases, an incorrect permission on a PHP and CGI script is to blame. These should usually be set at 0775 (-rwxr-xr-x).
      2. A PHP Timeout. If your script connects to external resources and those resources timeout, an HTTP 500 error can occur. Timeout rules, or better error handling in your script, should help if this is the cause of the 500 error.
      3. A Coding Error in .htaccess. While not as common, be sure to check that your site's .htaccess file is properly structured.
2. How would you edit or create a text file in linux/unix?
   1. If you meant using the command line, then you are asking how to create a file using the shell.
   2. cat >foo.txt
   3. touch foo.txt
3. There were standard java questions, like difference between final/finally/finalize
   1. Final: make variable unchangeable
      1. Make a method not over ridable
      2. Make a class not inheritable
   2. Finally:
      1. Try catch statement
   3. Finalize
      1. Called when object is garbage collected
4. Abstract class versus interface
   1. Abstract
      1. A Java abstract class is a class which cannot be instantiated, meaning you cannot create new instances of an abstract class. The purpose of an abstract class is to function as a base for subclasses.
   2. Interface
      1. A Java interface is a bit like a class, except a Java interface can only contain method signatures and fields. An Java interface cannot contain an implementation of the methods, only the signature (name, parameters and exceptions) of the method. You can use interfaces in Java as a way to achieve polymorphism.
5. thread.local, etc.
6. Also there were 5 or 6 exercises like merge two sorted linked lists as well as open ended questions like 'given the opportunity what would you change about the language or JVM'.
7. What is the memory management of java
8. how does garbage collection basically work?

The garbage collector is a program which runs on the Java Virtual Machine which gets rid of objects which are not being used by a Java application anymore. It is a form of automatic memory management.

When a typical Java application is running, it is creating new objects, such as Strings and Files, but after a certain time, those objects are not used anymore. For example, take a look at the following code:

for (File f : files) {

String s = f.getName();

}

In the above code, the String s is being created on each iteration of the for loop. This means that in every iteration, a little bit of memory is being allocated to make a String object.

Going back to the code, we can see that once a single iteration is executed, in the next iteration, the String object that was created in the previous iteration is not being used anymore -- that object is now considered "garbage".

Eventually, we'll start getting a lot of garbage, and memory will be used for objects which aren't being used anymore. If this keeps going on, eventually the Java Virtual Machine will run out of space to make new objects.

That's where the garbage collector steps in.

The garbage collector will look for objects which aren't being used anymore, and gets rid of them, freeing up the memory so other new objects can use that piece of memory.

In Java, memory management is taken care of by the garbage collector, but in other languages such as C, one needs to perform memory management on their own using functions such as malloc and free. Memory management is one of those things which are easy to make mistakes, which can lead to what are called memory leaks -- places where memory is not reclaimed when they are not in use anymore.

Automatic memory management schemes like garbage collection makes it so the programmer does not have to worry so much about memory management issues, so he or she can focus more on developing the applications they need to develop.

1. Describe scope and inheritance of variables.
2. Data Types
   1. Array- Value and key
   2. Stack- put in and remove from top
   3. Hash Table- In computing, a hash table (hash map) is a data structure used to implement an associative array, a structure that can map keys to values.
3. Data Sorting
   1. Bubble sort, sometimes referred to as sinking sort, is a simple sorting algorithm that repeatedly steps through the list to be sorted, compares each pair of adjacent items and swaps them if they are in the wrong order. The pass through the list is repeated until no swaps are needed, which indicates that the list is sorted. The algorithm, which is a comparison sort, is named for the way smaller elements "bubble" to the top of the list. Although the algorithm is simple, it is too slow and impractical for most problems even when compared to insertion sort.[1] It can be practical if the input is usually in sort order but may occasionally have some out-of-order elements nearly in position.
   2. Insertion Sort

**Insertion sort** is a simple [sorting algorithm](https://en.wikipedia.org/wiki/Sorting_algorithm) that builds the final [sorted array](https://en.wikipedia.org/wiki/Sorted_array) (or list) one item at a time. It is much less efficient on large lists than more advanced algorithms such as [quicksort](https://en.wikipedia.org/wiki/Quicksort), [heapsort](https://en.wikipedia.org/wiki/Heapsort), or [merge sort](https://en.wikipedia.org/wiki/Merge_sort). However, insertion sort provides several advantages:

* Simple implementation: [Bentley](https://en.wikipedia.org/wiki/Jon_Bentley) shows a three-line [C](https://en.wikipedia.org/wiki/C_%28programming_language%29) version, and a five-line optimized version[[1]](https://en.wikipedia.org/wiki/Insertion_sort#cite_note-pearls-1):116
* Efficient for (quite) small data sets, much like other quadratic sorting algorithms
* More efficient in practice than most other simple quadratic (i.e., [O](https://en.wikipedia.org/wiki/Big_O_notation)(*n*2)) algorithms such as [selection sort](https://en.wikipedia.org/wiki/Selection_sort) or [bubble sort](https://en.wikipedia.org/wiki/Bubble_sort)
* [Adaptive](https://en.wikipedia.org/wiki/Adaptive_sort), i.e., efficient for data sets that are already substantially sorted: the [time complexity](https://en.wikipedia.org/wiki/Time_complexity) is *O*(*nk*) when each element in the input is no more than *k* places away from its sorted position
* [Stable](https://en.wikipedia.org/wiki/Stable_sort); i.e., does not change the relative order of elements with equal keys
* [In-place](https://en.wikipedia.org/wiki/In-place_algorithm); i.e., only requires a constant amount O(1) of additional memory space
* [Online](https://en.wikipedia.org/wiki/Online_algorithm); i.e., can sort a list as it receives it

When people manually sort cards in a bridge hand, most use a method that is similar to insertion

1. The interview test covered MVC methodology via Struts 2, database interaction with
2. MySQL/JDBC, Java web technology like Tomcat and JSP, etc.
3. They asked a variant of the generic sorting algorithm question with keys and values but disallowed the use of a TreeMap (or any other additional data structure) in the solution.

As someone with a **Coach** WorkType, you are adept at growing relationships by helping people see themselves in a new light. Your support enables team members to develop new skills, discover new talents and push past their perceived limitations. You are attuned to people’s emotional needs and always available with a ready ear. You always seem to know just what to say. You track others’ progress and delight in their successes.

As someone with an **Explorer** WorkType, you are the creative spark and wellspring of ideas when things are just getting started. You see patterns and connections others don’t see. Your ability to keep an eagle’s eye perspective allows you to see new solutions to difficult problems. As a result, you are comfortable pushing into uncharted territory. Your vivid imagination and communication skills create a compelling vision that others naturally rally around.

**Jive Company**

**Vision Mission Etc**

Allow employees to work together without barriers

**Three Pillars Company Rebrand**

1. connect, communicate, collaborate

**CEO**

Elsia Steele

**Notes**

The most effective solutions are often focused on either enabling customer communities that include questions from popular social channels or developing self-service capabilities powered by a knowledge base. At Jive, we believe both elements are critically important, and that you shouldn’t choose one over the other.

**Products**

1. Jive-N
   1. Social collaboration software for employees
2. Jive-X
   1. Customer and partner communities
3. Jive Chime
   1. Real time messaging
4. Jive Circle
   1. Secure employer directory
5. Jive Daily
   1. Company news on the go

**Work Hard. Have Fun. Give Back.**

We work hard. We’re years ahead of the competition and we like keeping it that way. And we don’t just make and sell our solutions, we use them. Every day. To collaborate, ideate, imagine, argue, rescue, solve and create.

We have fun. We don’t take ourselves too seriously. And we like to celebrate. There’s always a happy hour, Green Machine Race or holiday gathering to take our minds off work.

And we give back. No matter how busy we are, we make time to contribute through volunteering and fundraising. Some of the causes we support include Adopt a Family, Walk for the Cure, food drives and more. We even give you paid time-off to volunteer for your favorite cause.

**Shared Mission**

To empower people and organizations to work better together.

**Manifesto**

We enable organizations to advance their mission by capturing the best of what every person has to offer. We enable people to do the best work of their lives with technology that adapts to their way – not the other way.

We make work more human and more productive.

**Values**

Collaboration

We support and inspire each other to do the best work of our lives.

Innovation

We tirelessly explore new ideas. We refuse to accept the status quo. We are creating the future.

Results

We help our customers and their communities achieve results that exceed their expectations.

Integrity

We are honest and transparent in all we do.

Fun

We are transforming an industry and having a great time doing it.

**Mission**

We’re driven by the evolution of the world. A fast-paced world where brilliant ideas come from anyone, anywhere, at any time. A world where anything is possible.

We enable organizations to advance their mission by capturing the best of what every person has to offer. We enable people to do the best work of their lives with technology that adapts to their way – not the other way.

Work has become less formal. From how we dress to how we communicate, collaborate and organize. Important decisions are now made in posts, in the halls, from our smartphones. People bring their technology habits and expectations from their homes into their work – and it just has to work.

The pace and diversity of work today makes it more important than ever for organizations to connect people, insights, ideas, information, opinions and creativity. To have a common purpose, a collective sense of direction. A sense of culture. A sense of community.

Where others see forums and posts, streams and likes – we see people. Generous, smart people helping each other in new ways so they can do their best and be their best. People with something to contribute, who relish the power of human connection, communication and collaboration.

We make work more human and more productive.

We make Monday mornings just a little bit brighter.

**We are Jive.**

**Interview**

**Story to draw on**

1. Taught Radiation Biology no previous college biology
   1. STAR
   2. Situation
      1. Got to teach Radiation Biology no previous college biology
2. Wishlist
3. Shareshare
4. File Share
5. Radiation Physics Thesis
6. Departmental Analysis
   1. Java Python
   2. Cross Department
   3. Fixed Staff Drive
7. New tools for graphic designers
   1. Put budget together
   2. Reasoning
   3. Sent to dean
   4. Approved
   5. Started marketing and graphic design office
8. Re did OSU Staff Drive

**Strengths**

Analytical Problem Solving Experience

* How to model radiation pulses

**Weaknesses**

Need to be more formal

**Interview Questions**

1. If someone called and complained that their Jive application was "slow" what are five initial questions you would ask?
2. What is the memory management of java and how does garbage collection basically work?
3. Identify the differences between two database engine configuration parameters which affect performance. Clearly state the limitations and differences between the two different database engine types
4. DNS issues usually result in what? How would you edit or create a text file in linux/unix? How would write a SQL SELECT statement? How would you handle an angry customer? Tell me about a time when you made a mistake, and how did you handle and/or resolve the issue?
5. Describe for us your process for working through a customer issue resolution cycle and your approach to customer communication
6. How would you troubleshoot a 500 Internal Server error? How would you deal with a customer who is not happy with your resolution to a problem? How do you deal with conflict? Tell me about a time when you couldn't resolve a customer's issue. How would you write a SQL SELECT statement? What are your hobbies, what do you do in your free time?
7. Write a SQL Join
8. Who is a role model/mentor of yours and how would they describe you?
9. 1 initial phone screening interview from HR - around 30 minutes - submission of one 'struts 2' web application (entailed learning new framework), total of around 32 hours of time spent learning the framework, developing program, and writing the user-instructions - 2 telephone screening interviews with Jive developers - around 30 minutes each - 5 in-person 1:1 interviews with different Jive employees (3 hours of straight interviewing) ---> After 8 interviews and a program submitted quickly and with zero bugs that they could find I was told that I came in a "close 2nd" and "thanks for your effort". One interviewer mentioned he preferred it when people submitted programs with bugs, so that they could ask them and ask about the bug and how it could be fixed. In my humble opinion requesting an applicant to learn a new framework and write a full blown web-application BEFORE doing the phone screening interviews is not very considerate. With that said though, I did genuinely like every person at Jive that I interacted with. This was all just very very disappointing for me.

**Technology**

**Tomcat Server**

**Maven**

**Questions**

There were standard java questions, like difference between final/finally/finalize, abstract class versus interface, thread.local, etc. Also there were 5 or 6 exercises like merge two sorted linked lists as well as open ended questions like 'given the opportunity what would you change about the language or JVM'. There were some standard db/sql questions such as differentiate between inner/left outer joins, indexes, etc as well as open ended questions such as tuning strategy/methodology for a slow application query, etc

**Linux Bash**

pwd (print working directory), cd (change directory), and ls (list files and directories)

**Jive Website**

**Combining Online Communities and Self-Service**

Customer support and success professionals face the steep challenge of meeting real-time needs in the age of the customer. In this collection, you’ll find valuable resources that cover best practices for measuring case deflection, knowledge base implementation, increasing customer satisfaction through community, and more.

**Combining the Best of Intranets and Enterprise Social Networks**

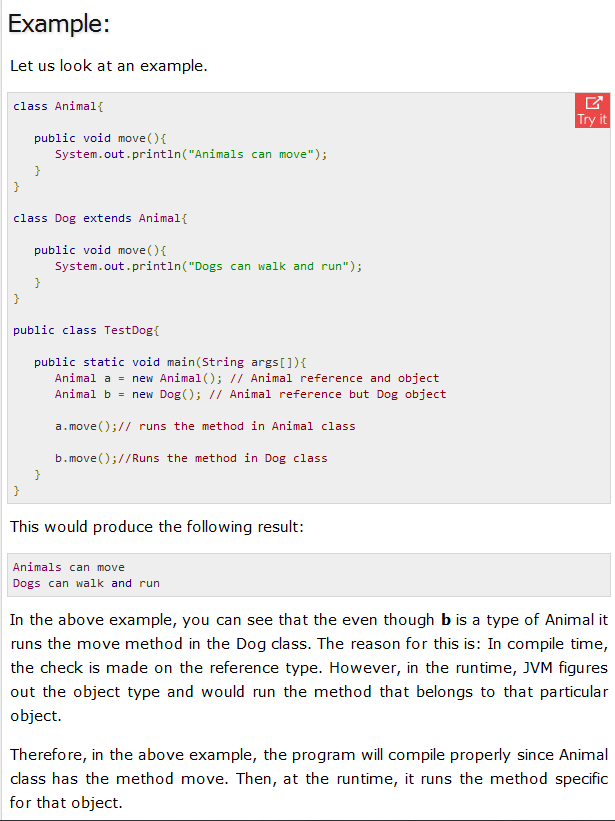
Today’s IT professionals are under pressure to meet the needs of business users faster than existing tools can deliver. In this collection, you’ll learn why intranets need to evolve – and how new “interactive intranets” reduce IT spend and drive higher performance for the entire organization.

**Redefining the CIO**

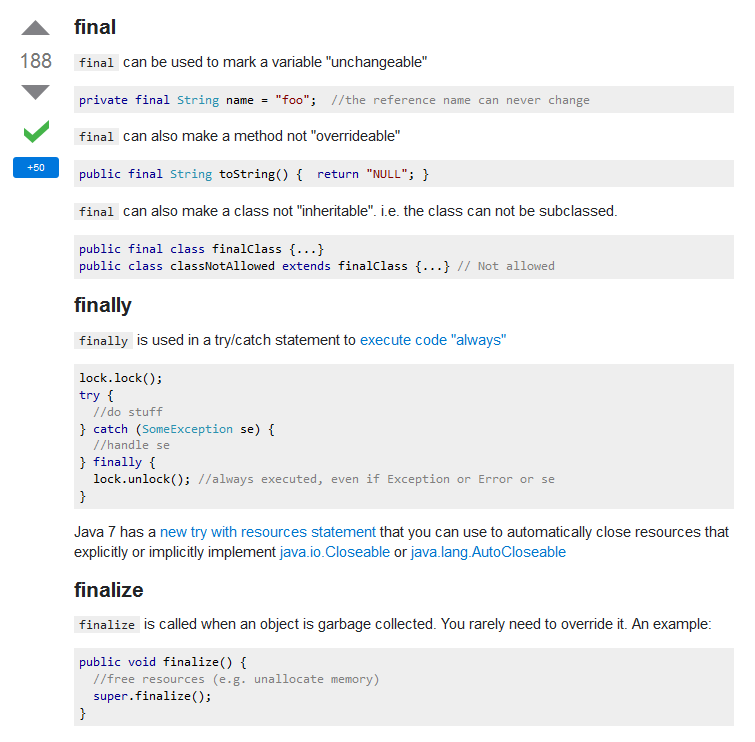
The job of the CIO is changing. Trends like cloud computing, the consumerization of IT and employees’ unprecedented access to technology are circumventing traditional controls and redefining the CIO’s role. Take a deep dive into the challenges and opportunities facing today’s IT leaders.

**Java**

**Overriding**



**Final Finally Finalized**



|  |  |  |
| --- | --- | --- |
| 3225 down vote accepted | The stack is the memory set aside as scratch space for a thread of execution. When a function is called, a block is reserved on the top of the stack for local variables and some bookkeeping data. When that function returns, the block becomes unused and can be used the next time a function is called. The stack is always reserved in a LIFO (last in first out) order; the most recently reserved block is always the next block to be freed. This makes it really simple to keep track of the stack; freeing a block from the stack is nothing more than adjusting one pointer.  The heap is memory set aside for dynamic allocation. Unlike the stack, there's no enforced pattern to the allocation and deallocation of blocks from the heap; you can allocate a block at any time and free it at any time. This makes it much more complex to keep track of which parts of the heap are allocated or free at any given time; there are many custom heap allocators available to tune heap performance for different usage patterns.  Each thread gets a stack, while there's typically only one heap for the application (although it isn't uncommon to have multiple heaps for different types of allocation).  To answer your questions directly:  *To what extent are they controlled by the OS or language runtime?*  The OS allocates the stack for each system-level thread when the thread is created. Typically the OS is called by the language runtime to allocate the heap for the application.  *What is their scope?*  The stack is attached to a thread, so when the thread exits the stack is reclaimed. The heap is typically allocated at application startup by the runtime, and is reclaimed when the application (technically process) exits.  *What determines the size of each of them?*  The size of the stack is set when a thread is created. The size of the heap is set on application startup, but can grow as space is needed (the allocator requests more memory from the operating system).  *What makes one faster?*  The stack is faster because the access pattern makes it trivial to allocate and deallocate memory from it (a pointer/integer is simply incremented or decremented), while the heap has much more complex bookkeeping involved in an allocation or free. Also, each byte in the stack tends to be reused very frequently which means it tends to be mapped to the processor's cache, making it very fast. Another performance hit for the heap is that the heap, being mostly a global resource, typically has to be multi-threading safe, i.e. each allocation and deallocation needs to be - typically - synchronized with "all" other heap accesses in the program.  A clear demonstration:  http://i.stack.imgur.com/i6k0Z.png Image source: [vikashazrati.wordpress.com](http://vikashazrati.wordpress.com/2007/10/01/quicktip-java-basics-stack-and-heap/)   |  | | --- | | [share](http://stackoverflow.com/a/80113)[improve this answer](http://stackoverflow.com/posts/80113/edit) | |

**Software Testing**

1. meets the requirements that guided its design and development,responds correctly to all kinds of inputs,
2. performs its functions within an acceptable time,
3. is sufficiently usable,
4. can be installed and run in its intended environments, and
5. achieves the general result its stakeholders desire.

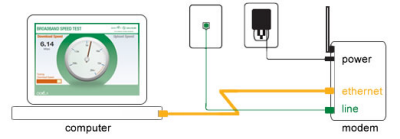
**How to troubleshoot a slow Internet connection**

Slow internet browsing can be very frustrating. It's possible that there is something specific to your location and equipment that is causing the slow down.

**Check your speed**

To figure out whether the lag time you're experiencing is too much:

1. Connect a computer to the modem with an [Ethernet cable](https://www.google.com/search?tbm=isch&q=ethernet+cable&oq=ethernet+cable).



1. Disconnect and power down all devices that access the Internet like other computers, gaming systems, Netflix or other movie streaming devices, DVRs, other routers, switches, VoIP phones, mobile phones and wireless printers.
2. Next, run a speed test and use the table below to figure out if you're getting the speed you should.
   * [CenturyLink Speed Test Tool](http://speedtest.centurylink.com)
   * [Other speed test tools](https://www.google.com/#q=test+my+internet+speed)

Note: Your speed can be affected by changes in network traffic and usage, which will usually vary across days and the time of day. To get the most accurate reading, run the speed test more than once, including different times of the day and different days of the week.

1. Write down the date, time and your results each time you run a test.
2. Compare your results against the table below. Are your speeds lower than or at/above the Target Download Speed?

|  |  |
| --- | --- |
| **Your Connection Speed** | **Target Download Speed (Wired)** |
| 1.5 Mbps | 1.2 Mbps |
| 3 Mbps | 2.4 Mbps |
| 5 Mbps | 4 Mbps |
| 7 Mbps | 5.6 Mbps |
| 12 Mbps | 9.6 Mbps |
| 20 Mbps | 16 Mbps |
| 40 Mbps | 32 Mbps |

1. If your speed test result was **at or above** the Target Download Speed, then you're receiving your subscribed speed. For example, if you've purchased a CenturyLink Internet package with 1.5 Mbps speed, your Target Download Speed shouldn't be lower than 1.2 Mbps.
2. To find out if faster speeds are available where you live:
   * Go to our [high-speed Internet services](http://www.centurylink.com/home/internet/) page.
   * Select the Get Internet Now button for what speeds are available in your area.
   * If faster speeds are not available, scroll to the bottom of the page and click the Notify Me button. We will email you when a faster speed is available where you live.

If the speed test shows your speed is **slower than** the Target Download Speed for your connection, you need to run a second test and check your modem speed.

* + Open up a web browser (Internet Explorer, Firefox, etc.).
  + Type <http://192.168.0.1>\* in the address bar, and hit Enter.
  + Select Modem Status and locate the rate (downstream/download or upstream/upload). If you don't see Modem Status, troubleshoot as if your speed test is less than 100%.

1. Still less than the Target Download Speed?
   * [Troubleshooting your modem -- check your cords and cables](http://www.centurylink.com/help/?assetid=192)
   * [Internet & Phone Working Together with Filters](http://www.centurylink.com/help/?assetid=185)
   * [Internet & Phone: How to properly connect a landline phone through the same wall phone jack as the modem](http://www.centurylink.com/help/?assetid=192#phonetomodem)

Lastly, bad wiring can cause slow connections. To figure out if the problem is inside or outside your house, [test your service at the Network Interface Device (NID)](http://internethelp.centurylink.com/internethelp/wired-dmarc.html).

**Still haven't found the problem?**

If your modem, filters and cables are installed correctly, it's possible your web browser isn't working superiorly, your wireless connection is partially blocked, a virus has found its way onto your computer and/or equipment is malfunctioning.

* If you haven't cleared your cache lately, your web browser may be bogged down and [cleaning it out](http://www.centurylink.com/help/?assetid=242#webbrowser) might speed up your connection.
* If you're using a wireless network, read [Improve the performance of your wireless connection](http://www.centurylink.com/help/?assetid=190) for quick troubleshooting ideas.
* If you have any reason to believe you have a virus, read [Keep your computer healthy with PC diagnostic and security scans](http://www.centurylink.com/help/?assetid=252) for how to check for viruses and how to remove them.
* If your modem is the oldest piece of networking equipment you have, it might be time to change it out. Old modems can slow down your connection speed.

If none of these things resonate with you, a [PC Health Check](https://centurylinkrc.com/AtEase/)\* might reveal some useful information. There is no cost to run a health check on your computer. The service is secure and takes about 7 minutes to complete. While the service is running, you can chat with an agent at any time by opening the LogMeInRescue window. When the health check is finished, you'll get a list of recommendations for what you can do next.

If you'd prefer to simply ask someone some questions, [chat with us](http://www.centurylink.com/help/chat/?team1=tech). We're here to help.

**Did you know?**

* The way a website is built or the equipment the website owner uses has a direct affect on the speed with which you experience a particular website or web page.
* Also, the amount of traffic on the internet can cause the speed of your service to vary. This is similar to a freeway you drive your car on. You may have a very fast car, but if the traffic on the freeway is slow, it is going to affect how fast your car can go. It has the same affect on your browsing experience.

Domain Name Servers (**DNS**) are the Internet's equivalent of a phone book. They maintain a directory of domain names and translate them to Internet Protocol (IP) addresses. This is necessary because, although domain names are easy for people to remember, computers or machines, access websites based on IP addresses.

The 500 Internal Server Error is a very general [HTTP status code](http://pcsupport.about.com/od/termshm/g/httpstatuscode.htm) that means something has gone wrong on the web site's server, but the server could not be more specific on what the exact problem is.

**Are You the Webmaster?** See Fixing 500 Internal Server Error Problems on Your Own Site towards the bottom of the page for some better advice if you're seeing the 500 Internal Server Error on one or more of your own pages.

The 500 Internal Server Error message might be seen in any number of ways because each website is allowed to customize the message.

Here are several common ways that you might see the HTTP 500 error:

### How You Might See a 500 Error

* **"500 Internal Server Error"**
* **"HTTP 500 - Internal Server Error"**
* **"Temporary Error (500)"**
* **"Internal Server Error"**
* **"HTTP 500 Internal Error"**
* **"500 Error"**
* **"HTTP Error 500"**
* **"500. That's an error"**

Since a 500 Internal Server Error is generated by the website you're visiting, you could see one in any browser in any [operating system](http://pcsupport.about.com/od/termshm/g/term_os.htm), even on your smartphone.

Most of the time, a 500 Internal Server Error displays inside the Internet browser window, just as web pages do.

continue reading below our video

How to Fix Browser Error Codes

### Cause of HTTP 500 Errors

As I mentioned above, Internal Server Error messages indicate that something, in general, is wrong.

Most of the time, "wrong" means an issue with the page or site's programming, but there's certainly a chance the problem is on your end, something we'll investigate below.

**Note:** More specific information about the cause of a particular HTTP 500 error is often provided when it occurs on a server using Microsoft IIS software.

Look for numbers after 500 as in **HTTP Error 500.19 - Internal Server Error** which means **Configuration data is invalid**. See More Ways You Might See an Internal Server Error below for the complete list.

### How To Fix the 500 Internal Server Error

Like I alluded to above, the 500 Internal Server Error is a server-side error, meaning the problem probably isn't with your computer or Internet connection but instead is a problem with the web site's server.

While not probable, it is possible that there's something wrong on your end and we'll look at some things you can try.

1. Reload the web page. You can do that by clicking the refresh/reload button, pressing **F5**, or trying the [URL](http://pcsupport.about.com/od/termsu/g/termurl.htm) again from the address bar.  
     
   Even if the 500 Internal Server Error is a problem on the web server, the issue may only be temporary. Trying the page again will often be successful.  
     
   **Note:** If the 500 Internal Server Error message appears during the checkout process at an online merchant, be aware that duplicate attempts to checkout may end up creating multiple orders - and multiple charges! Most merchants have automatic protections from these kinds of actions, but it's still something to keep in mind.

[Ads](http://pcsupport.about.com/od/findbyerrormessage/a/500servererror.htm)

* [PC Browser](http://pcsupport.about.com/z/js/o.htm?k=pc%20browser&SUName=pcsupport&d=PC%20Browser&r=http%3A%2F%2Fpcsupport.about.com%2Fod%2Ffindbyerrormessage%2Fa%2F500servererror.htm)
* [PC Server](http://pcsupport.about.com/z/js/o.htm?k=pc%20server&SUName=pcsupport&d=PC%20Server&r=http%3A%2F%2Fpcsupport.about.com%2Fod%2Ffindbyerrormessage%2Fa%2F500servererror.htm)
* [502 Error](http://pcsupport.about.com/z/js/o.htm?k=502%20error&SUName=pcsupport&d=502%20Error&r=http%3A%2F%2Fpcsupport.about.com%2Fod%2Ffindbyerrormessage%2Fa%2F500servererror.htm)
* [Managed Server](http://pcsupport.about.com/z/js/o.htm?k=managed%20server&SUName=pcsupport&d=Managed%20Server&r=http%3A%2F%2Fpcsupport.about.com%2Fod%2Ffindbyerrormessage%2Fa%2F500servererror.htm)
* [PC Help](http://pcsupport.about.com/z/js/o.htm?k=pc%20help&SUName=pcsupport&d=PC%20Help&r=http%3A%2F%2Fpcsupport.about.com%2Fod%2Ffindbyerrormessage%2Fa%2F500servererror.htm)

1. [Clear your browser's cache](http://pcsupport.about.com/od/browsers/f/clear-cache.htm). If there's a problem with the cached version of the page you're viewing, it could be causing HTTP 500 issues.  
     
   **Note:** Internal Server Errors are not often caused by caching issues, but I have, on occasion, seen the error go away after clearing the cache. It's such an easy and harmless thing to try so don't skip it.
2. [Delete your browser's cookies](http://pcsupport.about.com/od/browsers/f/delete-cookies.htm). Some 500 Internal Server Error issues can be corrected by deleting the cookies associated with the site you're getting the error on.  
     
   After removing the cookie(s), restart the browser and try again.
3. Troubleshoot as a [504 Gateway Timeout](http://pcsupport.about.com/od/findbyerrormessage/a/504error.htm) error instead.  
     
   It's not very common, but some servers produce a 500 Internal Server Error when in reality a more appropriate message based on the cause of the problem is 504 Gateway Timeout.
4. Contacting the website directly is another option. Chances are good the site's administrators already know about the 500 error but if you suspect they don't, letting them know helps both you and them (and everyone else).  
     
   See my [Website Contact Information](http://pcsupport.about.com/od/resources/tp/website-contact-information.htm) list for contact information for popular websites. Most sites have support-based social network accounts and a few even have email and telephone numbers.  
     
   **Tip:** If it looks like the site is down completely and you can't find a way to report the 500 Internal Server Error message to the website, it might help your sanity to keep up with the outage on Twitter. You can usually do this by searching for #websitedown on Twitter, as in [#gmaildown](https://twitter.com/search?q=%23gmaildown) or [#facebookdown](https://twitter.com/search?q=%23facebookdown).
5. Come back later. Unfortunately, at this point, the 500 Internal Server Error is no doubt a problem outside your control that will eventually get fixed by someone else.  
     
   If the 500 Internal Server Error message is appearing at check out during an online purchase, it might help to realize that sales are probably being disrupted - usually a great incentive to the online store to fix the issue very quickly!  
     
   Even if you're getting the 500 error on a site that doesn't sell anything, like YouTube or Twitter, as long as you've let them know about the problem, or at least tried, there's little more you can do than wait it out.

### Fixing 500 Internal Server Error Problems on Your Own Site

A 500 Internal Server Error on your own website requires a completely different course of action. As I mentioned above, most 500 errors are server-side errors, meaning it's likely your problem to fix if it's your website.

There are lots of reasons why your site might be serving a 500 Error to your users, but two are most common:

* **A Permissions Error.** In most cases, a 500 Internal Server Error is due to an incorrect permission on one or more files or folders. In most of those cases, an incorrect permission on a PHP and CGI script is to blame. These should usually be set at 0775 (-rwxr-xr-x).
* **A PHP Timeout.** If your script connects to external resources and those resources timeout, an HTTP 500 error can occur. Timeout rules, or better error handling in your script, should help if this is the cause of the 500 error.
* **A Coding Error in .htaccess.** While not as common, be sure to check that your site's [.htaccess file](http://pcsupport.about.com/od/fileextensions/f/htaccess-file.htm) is properly structured.

If you're running [WordPress](https://wordpress.org/support/), [Joomla](http://www.joomla.org/), or another content management or CMS system, be sure to search their support centers for more specific help troubleshooting a 500 Internal Server Error.

If you're not using an off-the-shelf content management tool, your web hosting provider, like [InMotion](http://www.inmotionhosting.com/support/website/error-numbers/500-internal-server-error), [Dreamhost](http://wiki.dreamhost.com/Advanced_Troubleshooting_Techniques), [Bluehost](https://my.bluehost.com/cgi/help/594), [1&1](http://help.1and1.com/hosting-c37630/scripts-and-programming-languages-c85099/php-c37728/explanation-of-500-errors-a595888.html), etc., probably has some 500 Error help that might be more specific to your situation.

### More Ways You Might See an Internal Server Error

In Internet Explorer, the message **The website cannot display the page** often indicates an HTTP 500 Internal Server Error. A 405 Method Not Allowed error is another possibility but you can be sure by looking for either 500 or 405 in the IE title bar.

When Google services, like Gmail or Google+, are experiencing a 500 Internal Server Error, they often report a Temporary Error (500) or simply 500.

When [Windows Update](http://pcsupport.about.com/od/keepingupwithupdates/p/windows-update.htm) reports an Internal Server Error, it appears as a WU\_E\_PT\_HTTP\_STATUS\_SERVER\_ERROR message or as the 0x8024401F error code.

If the website that reports the 500 error is running Microsoft IIS, you may get a more specific error message:

**Thread**

Multithreading refers to two or more tasks executing concurrently within a single program. A **thread** is an independent path of execution within a program. Many **threads** can run concurrently within a program. Every **thread** in **Java** is created and controlled by the **java**.lang.**Thread** class.

A terminal is at the end of an electric wire, a shell is the home of a turtle, tty is a strange abbreviation and a console is a kind of cabinet.

Well, etymologically speaking, anyway.

In unix terminology, the short answer is that

* terminal = tty = text input/output environment
* console = physical terminal
* shell = command line interpreter

Console, terminal and tty are closely related. Originally, they meant a piece of equipment through which you could interact with a computer: in the early days of unix, that meant a [teleprinter](http://en.wikipedia.org/wiki/Teleprinter)-style device resembling a typewriter, sometimes called a teletypewriter, or “tty” in shorthand. The name “terminal” came from the electronic point of view, and the name “console” from the furniture point of view. Very early in unix history, electronic keyboards and displays became the norm for terminals.

In unix terminology, a **tty** is a particular kind of [device file](http://en.wikipedia.org/wiki/Device_file) which implements a number of additional commands ([ioctls](http://en.wikipedia.org/wiki/Ioctl" \l "Terminals)) beyond read and write. In its most common meaning, **terminal** is synonymous with tty. Some ttys are provided by the kernel on behalf of a hardware device, for example with the input coming from the keyboard and the output going to a text mode screen, or with the input and output transmitted over a serial line. Other ttys, sometimes called **pseudo-ttys**, are provided (through a thin kernel layer) by programs called [**terminal emulators**](http://en.wikipedia.org/wiki/Terminal_emulator), such as [Xterm](http://en.wikipedia.org/wiki/Xterm) (running in the [X Window System](http://en.wikipedia.org/wiki/X_Window_System)), [Screen](http://en.wikipedia.org/wiki/Gnu_screen) (which provides a layer of isolation between a program and another terminal), [Ssh](http://en.wikipedia.org/wiki/Secure_shell) (which connects a terminal on one machine with programs on another machine), [Expect](http://en.wikipedia.org/wiki/Expect) (for scripting terminal interactions), etc.

The word terminal can also have a more traditional meaning of a device through which one interacts with a computer, typically with a keyboard and display. For example an X terminal is a kind of [thin client](http://en.wikipedia.org/wiki/Thin_client), a special-purpose computer whose only purpose is to drive a keyboard, display, mouse and occasionally other human interaction peripherals, with the actual applications running on another, more powerful computer.

A **console** is generally a terminal in the physical sense that is by some definition the primary terminal directly connected to a machine. The console appears to the operating system as a (kernel-implemented) tty. On some systems, such as Linux and FreeBSD, the console appears as several ttys (special key combinations switch between these ttys); just to confuse matters, the name given to each particular tty can be “console”, ”virtual console”, ”virtual terminal”, and other variations.

See also [Why is a Virtual Terminal “virtual”, and what/why/where is the “real” Terminal?](http://askubuntu.com/q/14284/1059).

A [**shell**](http://en.wikipedia.org/wiki/Shell_%28computing%29) is the primary interface that users see when they log in, whose primary purpose is to start other programs. (I don't know whether the original metaphor is that the shell is the home environment for the user, or that the shell is what other programs are running in.)

In unix circles, **shell** has specialized to mean a [command-line shell](http://en.wikipedia.org/wiki/Shell_%28computing%29#Text_.28CLI.29_shells), centered around entering the name of the application one wants to start, followed by the names of files or other objects that the application should act on, and pressing the Enter key. Other types of environments don't use the word “shell”; for example, window systems involve “[window managers](http://en.wikipedia.org/wiki/Window_manager)” and “[desktop environments](http://en.wikipedia.org/wiki/Desktop_environment)”, not a “shell”.

There are many different unix shells. Popular shells for interactive use include [Bash](http://en.wikipedia.org/wiki/Bash_%28Unix_shell%29) (the default on most Linux installations), [zsh](http://en.wikipedia.org/wiki/Zsh) (which emphasizes power and customizability) and [fish](http://en.wikipedia.org/wiki/Friendly_interactive_shell) (which emphasizes simplicity).

Command-line shells include flow control constructs to combine commands. In addition to typing commands at an interactive prompt, users can write scripts. The most common shells have a common syntax based on the [Bourne\_shell](http://en.wikipedia.org/wiki/Bourne_shell). When discussing “**shell programming**”, the shell is almost always implied to be a Bourne-style shell. Some shells that are often used for scripting but lack advanced interactive features include [the Korn shell (ksh)](http://en.wikipedia.org/wiki/Korn_shell) and many [ash](http://en.wikipedia.org/wiki/Almquist_shell) variants. Pretty much any Unix-like system has a Bourne-style shell installed as /bin/sh, usually ash, ksh or bash.

In unix system administration, a user's **shell** is the program that is invoked when they log in. Normal user accounts have a command-line shell, but users with restricted access may have a [restricted shell](http://en.wikipedia.org/wiki/Restricted_shell) or some other specific command (e.g. for file-transfer-only accounts).

The division of labor between the terminal and the shell is not completely obvious. Here are their main tasks.

* Input: the terminal converts keys into control sequences (e.g. Left → \e[D). The shell converts control sequences into commands (e.g. \e[D → backward-char).
* Line edition, input history and completion are provided by the shell.
  + The terminal may provide its own line edition, history and completion instead, and only send a line to the shell when it's ready to be executed. The only common terminal that operates in this way is M-x shell in Emacs.
* Output: the shell emits instructions such as “display foo”, “switch the foreground color to green”, “move the cursor to the next line”, etc. The terminal acts on these instructions.
* The prompt is purely a shell concept.
* The shell never sees the output of the commands it runs (unless redirected). Output history (scrollback) is purely a terminal concept.
* Inter-application copy-paste is provided by the terminal (usually with the mouse or key sequences such as Ctrl+Shift+V or Shift+Insert). The shell may have its own internal copy-paste mechanism as well (e.g. Meta+W and Ctrl+Y).
* [Job control](http://en.wikipedia.org/wiki/Job_control) (launching programs in the background and managing them) is mostly performed by the shell. However, it's the terminal that handles key combinations like Ctrl+C to kill the foreground job and Ctrl+Z to suspend it.

# What Is an Interface?

As you've already learned, objects define their interaction with the outside world through the methods that they expose. Methods form the object's *interface* with the outside world; the buttons on the front of your television set, for example, are the interface between you and the electrical wiring on the other side of its plastic casing. You press the "power" button to turn the television on and off.

In its most common form, an interface is a group of related methods with empty bodies. A bicycle's behavior, if specified as an interface, might appear as follows:

interface Bicycle {

// wheel revolutions per minute

void changeCadence(int newValue);

void changeGear(int newValue);

void speedUp(int increment);

void applyBrakes(int decrement);

}

To implement this interface, the name of your class would change (to a particular brand of bicycle, for example, such as ACMEBicycle), and you'd use the implements keyword in the class declaration:

class ACMEBicycle **implements** Bicycle {

int cadence = 0;

int speed = 0;

int gear = 1;

// The compiler will now require that methods

// changeCadence, changeGear, speedUp, and applyBrakes

// all be implemented. Compilation will fail if those

// methods are missing from this class.

void changeCadence(int newValue) {

cadence = newValue;

}

void changeGear(int newValue) {

gear = newValue;

}

void speedUp(int increment) {

speed = speed + increment;

}

void applyBrakes(int decrement) {

speed = speed - decrement;

}

void printStates() {

System.out.println("cadence:" +

cadence + " speed:" +

speed + " gear:" + gear);

}

}

Implementing an interface allows a class to become more formal about the behavior it promises to provide. Interfaces form a contract between the class and the outside world, and this contract is enforced at build time by the compiler. If your class claims to implement an interface, all methods defined by that interface must appear in its source code before the class will successfully compile.