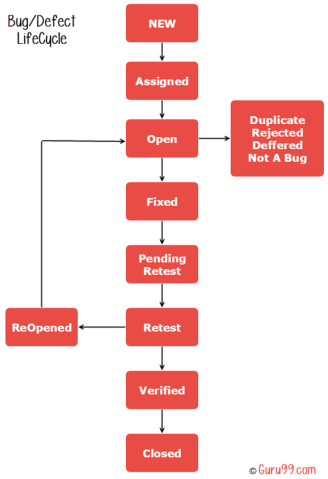
1. Difference between Priority and Severity

SN	Priority	Severity
1.	Defect Priority has defined the order in which the developer should resolve a defect	Defect Saverity is defined as the degree of impact that a defect on the operation of the product.
2.	Priority is associated with scheduling	Severity is associated with functionality or standards
3.	Priority indicates how soon the bug should be fixed	Severity indicates the seriousness of the defect on he product functionality
4.	Priority of defects is decided in consultation with the manager/client	QA engineer determines the severity level of the defect.
5.	Priority is driven by business value	Severity is driven by functionality
6.	Its value is subjective	Its value is objective
7.	Priority is of 3 types: Low, Medium, and High.	Severity is of 5 types: Critical, Major, Moderate, Minor, and Cosmetic.

2. What is Bug Life cycle?
   - "A computer bug is an error, flaw, mistake, failure, or fault in a computer program that prevents it from working correctly or produces an incorrect result. Bugs arise from mistakes and errors, made by people, in either a program's source code or its design."
   - The duration or time span between the first time defects is found and the time that it is closed successfully, rejected, postponed or deferred is called as 'Defect Life Cycle'.



   - The above diagram shows different states of Defect in Defect Life Cycle and these are as follows:

1. NEW - When a new defect is logged and posted for the first time. It is assigned a status as NEW.
2. ASSIGNED - Once the bug is posted by the tester, the lead of the tester approves the bug and assigns the bug to the developer team.
3. Open: The developer starts analyzing and works on the defect fix
4. Fixed: When a developer makes a necessary code change and verifies the change, he or she can make bug status as "Fixed."
5. Pending retest: Once the defect is fixed the developer gives a particular code for retesting the code to the tester. Since the software testing remains pending from the testers end, the status assigned is "pending retest."

6. Retest: Tester does the retesting of the code at this stage to check whether the defect is fixed by the developer or not and changes the status to “Re-test.”
7. Verified: The tester re-tests the bug after it got fixed by the developer. If there is no bug detected in the software, then the bug is fixed and the status assigned is “verified.”
8. Reopen: If the bug persists even after the developer has fixed the bug, the tester changes the status to “reopened”. Once again the bug goes through the life cycle.
9. Closed: If the bug is no longer exists then tester assigns the status “Closed.”
10. Duplicate: If the defect is repeated twice or the defect corresponds to the same concept of the bug, the status is changed to “duplicate.”
11. Rejected: If the developer feels the defect is not a genuine defect then it changes the defect to “rejected.”
12. Deferred: If the present bug is not of a prime priority and if it is expected to get fixed in the next release, then status “Deferred” is assigned to such bugs
13. Not a bug:If it does not affect the functionality of the application then the status assigned to a bug is “Not a bug”.

3. What is priority?
   - Priority defines the order in which we should resolve a defect. Should we fix it now, or can it wait? This priority status is set by the tester to the developer mentioning the time frame to fix the defect. If high priority is mentioned then the developer has to fix it at the earliest. The priority status is set based on the customer requirements.
   - For example: If the company name is misspelled in the home page of the website, then the priority is high and severity is low to fix it. Priority can be of following types:
     - Low: The defect is an irritant which should be repaired, but repair can be deferred until after more serious defect has been fixed.
     - Medium: The defect should be resolved in the normal course of development activities. It can wait until a new build or version is created.
     - High: The defect must be resolved as soon as possible because the defect is affecting the application or the product severely. The system cannot be used until the repair has been done.
     - Critical: Extremely urgent, resolve immediately
4. What is severity?
   - Severity is absolute and Customer-Focused. It is the extent to which the defect can affect the software. In other words, it defines the impact that a given defect has on the system.
     - For example: If an application or web page crashes when a remote link is clicked, in this case clicking the remote link by an user is rare but the impact of application crashing is severe. So the severity is high but priority is low
   - Severity can be of following types:
     - Critical: The defect that results in the termination of the complete system or one or more component of the system and causes extensive corruption of the data. The failed function is unusable and there is no acceptable alternative method to achieve the required results then the severity will be stated as critical
     - Major (High): The defect that results in the termination of the complete system or one or more component of the system and causes extensive corruption of the data. The failed function is unusable but there exists an acceptable alternative method to achieve the required results then the severity will be stated as major.

- Moderate (Medium): The defect that does not result in the termination, but causes the system to produce incorrect, incomplete or inconsistent results then the severity will be stated as moderate.
- Minor (Low): The defect that does not result in the termination and does not damage the usability of the system and the desired results can be easily obtained by working around the defects then the severity is stated as minor.
- Cosmetic: The defect that is related to the enhancement of the system where the changes are related to the look and field of the application then the severity is stated as cosmetic.

5. Bug categories are…

- Bug categories:- security, database, functionality (critical/general), UI

6. Advantage of Bugzila .

- Bugzilla is open source issue/bug tracking system.
- Bugzilla is defect tracking tools.
- This open bug tracker enables users stay connected with their clients, employees to communicate about problems effectively throughout data management tools.
- Advanced search capabilities
- E-mail notifications
- Modify/file bug by e-mail
- Time tracking, strong security, customization, localization.