1. **Mention what are the categories of defects?**

* There are different categories of defect

1. Logical defect
2. Arithmetic defect
3. Syntax defect
4. Multithreading defect
5. Interface defect
6. Performance defect
7. Software error
8. Software fault
9. Software failure
10. Boundary and range defect
11. Data validation Defect
12. Deployment defect
13. Integration defect
14. Documentation defect
15. Data defect
16. Security defect
17. User interface defect
18. **Difference between priority and severity**

|  |  |
| --- | --- |
| **Priority Defect** | **Severity Defect** |
| Priority determines the defect urgency of repair. | Severity determines the defect effect on the application. |
| **How bad the defect is -** Severity is given by QA tester | **How soon we need to fix -** priority is given by Test Lead or project manager |
| Priority determined by business value and impact on customer | Severity determined by the app performance and functionality |
| Level:- **Critical** – The software will not run **High** – Unexpected fatal error( including crash and data corruption) **Medium** – A feature is malfunctioning  **Low** – A cosmetic issue | Level:-  P1 – Fix before next build to test  P2 – Fix before final release  P3 – We probably won’t to these, but we want to track them anyway to resolve the priority severity divide. |
| It is associated with scheduling | It is associated with standard or high principles. |
| A high priority of bug marked for a fix immediately. | A high severity of bug marked for fix immediately or later. |

1. **What is Bug life Cycle**

* The duration or time span between the first-time defect is found and the time that it is closed successfully, rejected, postponed or deferred is called bug(defect) life cycle.
* **Here some phase of bug life cycle**
* 

1. **New**
2. **Assigned**
3. **Open**
4. **Fixed**
5. **Pending retest**
6. **Retest**
7. **Reopen**
8. **Duplicated**
9. **Rejected**
10. **Deffered**
11. **Not a bug**
12. **Verifies**
13. **Closed**
14. **New:-** When a new defect is logged and posted for the first time. It is assigned a status as NEW.
15. **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.
16. **Open:-** The developer starts analyzing and works on the defect fix
17. **Fixed:-** When a developer makes a necessary code change and verifies the change, he or she can make bug status as “Fixed.”
18. **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.”
19. **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.”
20. **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.”
21. **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.
22. **Closed:-** If the bug is no longer exists then tester assigns the status “Closed.”
23. **Duplicate:-** If the defect is repeated twice or the defect corresponds to the same concept of the bug, the status is changed to “duplicate.”
24. **Rejected:-** If the developer feels the defect is not a genuine defect then it changes the defect to “rejected.”
25. **Deffered:-** 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
26. **Not a bug:-** If it does not affect the functionality of the application then the status assigned to a bug is “Not a bug”.
27. **What is priority?**

* Defect priority is a measure of how important and urgent a defect is, and how soon it needs to be fixed.
* If high priority is mentioned then the developer has to fix at the earlier. The priority status is set based on the customer requirement.

**For example:-** if the company name mistake 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:-

1. **Low:-**

A defect that does not have any major impact on the functionality of the software and hence does not need any immediate attention.

**-** It can be repaired in the future or once higher-priority defects are fixed.

**-** All the severity defects fall into this category.

**-** It is represented by p4.

1. **Medium:-**

A defect with minor severity that need not be fixed right away as it does not cause any significant functionality issues or business.

* These defect are fixed after immediate and high-priority defects are removed.
* All minor severity defects fall into this category.
* It is represented by p3.

1. **High:-**

* The defect comes immediately as it needs less attention but still more than other remaining priority categories.
* The defect must be resolve 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.

1. **Critical:-**

* Extremely urgent, resolve immediately.

**5.What is severity?**

- Defect severity measure how much a defect impacts a software system’s functionality for end-users. It is also known as bug severity.

**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 following types:-

1. Critical:- A defect that leads to the complete failure of the software system.

For example, a system that crashes, closes abruptly, or corrupts data.

1. Major:- A defect that leads to the failure of a crucial part of the application.
2. Moderate:- A defect that has a noticeable impact but is not as critical as high severity defects.
3. Minor:- A defect that is a minor issue that does not significantly affect the overall functionality of the features.

**6.Bug categories are……**

**-** There are different categories of bug:-

1. Security bug

2. Functional bug

3. Usability bug

4. Performance bug

5. Unit level bug

6. critical functionality bug

7. Logical bug

8. User interface bug

**7. Advantage of Bugzilla.**

- Bugzilla is a free, web-based defect-tracking system that helps teams manage bugs, issues, and other change requests.

- It has many advantages, including: -

**Customizable workflows: -** Bugzilla allows users to create custom workflows for reporting, prioritizing, and fixing bugs.

**Notifications: -** Bugzilla can send email notifications when changes are made to bugs. Users can control which bugs they receive notifications for.

**Improved product quality: -** Bugzilla can help improve the quality of products by streamlining the process or reporting, prioritizing, and fixing bugs.