1.What Is git?What is gt checkout?Explan about merging and patch creation?

Ans: As **Git** is a distributed version control system, it can be used as a server out of the box. Dedicated **Git** server software helps, amongst other features, to add access control, display the contents of a **Git** repository via the web, and help managing multiple repositories.



When ever we are working with git,we will create branches,to do our work,after our work completion we have to merge that branch with master branch.That will be happen by following command.

Git merge <branch name>

git format-patch master --stdout > fix\_empty\_poster.patch

This will create a new file fix\_empty\_poster.patch with all changes from the current (fix\_empty\_poster) against master. Normally, git would create a separate patch file for each commit, but that’s not what we want. All we need is a single patch file.

Now, you have a patch for the fix you wrote. Send it to the maintainer of the project …

2.Difference between git status and git log?

Ans: While **git status** lets you inspect the working directory and the staging area, **git log**only operates on the committed history. **Log** output can be customized in several ways, from simply filtering commits to displaying them in a completely user-defined format.

3.Version categarization of source code…….How will you do?

Ans: git tag -tagname

4.In master repository if different branches are created how will you differentiate the differnces b/w previous source code to latest source code?

Ans:git diff

5.What is integration?How will you integrate software code?

Ans: System **integration** is defined in engineering as the process of bringing together the component sub-systems into one system (an aggregation of subsystems cooperating so that the system is able to deliver the overarching functionality) and ensuring that the subsystems function together as a system, and in informatio

<https://en.wikipedia.org/wiki/System_integration>

If we are using version control tools to maintain a source code ,by using that we can integrate our source codes with each others source code.

6.What is perforce?

Ans: <https://www.perforce.com/perforce/doc.current/manuals/intro/01_intro.html>

7.what is label? What is the command for label and its use?How anyone will know what is the previous label of the file?

Ans: A Perforce *label* is a user-determined list of files and revisions. The label can later be used to reproduce the state of these files within a client workspace.

Labels provide a method of naming important combinations of file revisions for later reference. For example, the file revisions that comprise a particular release of your software might be given the label release2.0.1. At a later time, you can retrieve all the files in that label into a client workspace with a single command.

**p4 [*g-opts*] label -o [-t *template*] *labelname***

p4 labels <filename>

8.what is changelist?

Ans: A **Perforce changelist** is a list of files, their revision numbers, and operations to be performed on these files. You add files to a **changelist** with commands such as p4 add filenames or p4 edit filenames , and the changed files are stored in the depot when you submit the **changelist** with p4 submit

9.How will you push the test cases in capella tool?

Ans:

10.how to create branch in clearcase?

Ans: mkbranch

11.what is common view,user view and integrated view?

Ans:

12.What Is CR ?states of CR?

Ans: Internal **change requests** can involve a wide variety of actions including patching and software and hardware upgrades. Once a **change request** has been made, the process of change control should be undertaken to make sure that the request is satisfied efficiently and without unnecessary use of resources.

1. Step 1 – Determine the Scope of the Change. ...
2. Step 2 – Determine the Scope of Incorporating the Change. ...
3. Step 3 – Gain Approval or Rejection of the Change. ...
4. Step 4 – Communicate and Implement an Approved Change Request. ...

13.what is QXDM?uses of that?

Ans: The Qualcomm eXtensible Diagnostic Monitor Professional (**QXDM** Pro) tool provides a diagnostic client for rapid prototyping of new clients and protocol packets. It utilizes a graphical user interface to display data transmitted to and from the subscriber station.

**Use QXDM** Pro to capture and monitor Qualcomm proprietary diagnostic logs, messages, and events. ... **QXDM** supports all Qualcomm devices with enabled diagnostic services. A Qualcomm USB driver is required to establish the connection. Capture and analyze the diagnostic packets defined in the interface control document.

The QUALCOMM eXtensible Diagnostic Monitor (QXDMProfessional) is a real-time data collection and diagnosticlogging tool for measuring mobile-based RF performance.Designed to operate using all commercial handsets thatcontain QUALCOMM ASICs and with QUALCOMM’stest/trial phones,\* QXDM Professional displays statisticsand diagnostic information, and enables users to read andwrite non-volatile memory. Whether conducting tests inthe lab or the eld, QXDM Professional is a powerfulplatform for evaluating handset and network performance

14.What is QPST?When do we use QXDM?

Ans: QPST means Qualcomm product support tools. QPST and QXDM are used to  determine your cell phone's transmit and receive functions are working properly or not.

The QUALCOMM extensible Diagnostic Monitor (QXDM Professional) is a real-time data collection and diagnostic logging tool for measuring mobile-based RF performance. Designed to operate using all commercial handsets that contain QUALCOMM ASICs and with QUALCOMM’s test/trial phones,\* QXDM Professional displays statistics and diagnostic information, and enables users to read and write non-volatile memory. Whether conducting tests in the lab or the eld, QXDM Professional is a powerful platform for evaluating handset and network performance

15.What is protocol log analysis?

Ans:

16.How to create mirror build or temporary build?

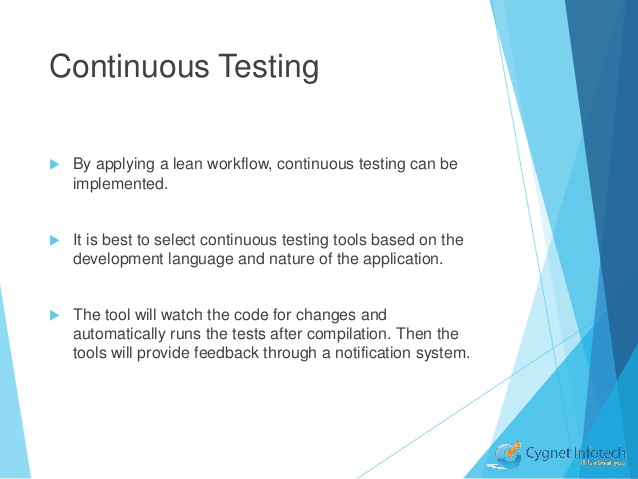
Ans:

17.What is ARIS?

Ans:

18.What is CIT? For which builds we ise cit?

Ans:CIT means Continous Integaration Testing.



19. .What is MTF? For which builds we use MTF?

Ans:MTF means Modulation Transfr function.

Quick MTF checks the quality of images taken with an imaging device, letting you test the device's performance in different conditions and settings. This enables you to make an informed decision about whether the device meets your needs.

Quick MTF can be used to check virtually any imaging equipment; it is successfully used for testing the quality of lenses, cameras, medical imaging devices, night vision systems, microscopes, surveillance systems, and so on.

20.How to fix the CR for a particular build?

Ans:

Create the change request with one of these options.

1. **From the Change module**

**2.From an incident or a problem.**

**3.From an existing change record.**

21.Explain about PRISM?

Ans:

22.What is Gerrit? When do u use it?

Ans: **Gerrit** is a free, web-based team code collaboration tool. Software developers in a team can review each other's modifications on their source code using a Web browser and approve or reject those changes. It integrates closely with Git, a distributed version control system.

23.What is SVN?

Ans: Apache Subversion which is often abbreviated as SVN, is a software versioning and revision control system distributed under an open source license. Subversion was created by CollabNet Inc. in 2000, but now it is developed as a project of the Apache Software Foundation, and as such is part of a rich community of developers and users. This tutorial provides you an understanding on SVN system that is needed to maintain the current and historical versions of files such as source code, web pages, and documentations.

24.Explain about jenkins?

Ans: **Jenkins** is an open-source continuous integration software tool written in the Java programming language for testing and reporting on isolated changes in a larger code base in real time. The software enables developers to find and solve defects in a code base rapidly and to automate testing of their builds.

Continuous integration is a process in which all development work is integrated as early as possible. The resulting artifacts are automatically created and tested. This process allows to identify errors as early as possible.

[Jenkins](https://jenkins.io/) is a popular open source tool to perform continuous integration and build automation. The basic functionality of Jenkins is to execute a predefined list of steps, e.g. to compile Java source code and build a JAR from the resulting classes. The trigger for this execution can be time or event based. For example, every 20 minutes or after a new commit in a Git repository.

25.How will integrate bluez driver ?

Ans:<http://opensourceforu.com/2015/06/linux-without-wires-the-basics-of-bluetooth/>

26.What is continuous integration?

Ans; Continuous Integration (CI) is a development practice that requires developers to integrate code into a shared repository several times a day. Each check-in is then verified by an automated build, allowing teams to detect problems early.   
  
By integrating regularly, you can detect errors quickly, and locate them more easily.

* Say goodbye to long and tense integrations
* Increase visibility enabling greater communication
* Catch issues early and nip them in the bud
* Spend less time debugging and more time adding features
* Build a solid foundation
* Stop waiting to find out if your code’s going to work
* Reduce integration problems allowing you to deliver software more rapidly

27.Jenkins installation in Ubuntu?

Ans; <https://wiki.jenkins.io/display/JENKINS/Installing+Jenkins+on+Ubuntu>

28.When merge conflicts occur how do u resolve?

Ans: <https://help.github.com/articles/resolving-a-merge-conflict-using-the-command-line/>