1. What is software life cycle?

Software life cycle includes various steps to design , develop and maintain software projects and ensures that a high quality, cost effective, reliable software is developed which satisfies all customer requirements. These SDLC model is made for specific type of project , we cannot deploy one single model for every project as every project has different requirements. Following are the phases of software life cycle.

Requirement analysis:
The first phase emphasis on gathering requirements from customers. We get to know what is needed from system to be built. The output of this document is SRS which gives comprehensive description about the behavior of software.

Design:
The aim of this phase is to transform the requirement from previous phase into plan. This means the software developers and designers define the plan to solution.

Coding:
In this phase the design is converted into code by using technical language. The output of this logical phase is program code.

Testing
Here the entire code and overall functionality of the software is tested to know whether the developed software works fine according to the requirements specified by customer. It is an important phase as effective testing will provide high quality products, lower maintenance cost and accurate results.

Maintenance:
At this phase , the software developed is delivered to end user who is responsible for proper function and maintenance.

2.What is waterfall software development and what are its major limitations?

It is a software development model which follows sequential flow of development, hence it is also called linear sequential model. As it follows sequential structure it is less iterative and easy to use compared to other models of development.

Flow of waterfall model is as follows:
1. System design
2. Implementation
3. Testing
4. Development
5. Maintenance

As each of the methodology stages is completed, the developers can move on to the next step. The key point about the Waterfall design methodology is that there's no chance for changes or errors, so your project outcome and a detailed plan must be set in the beginning.

Disadvantages:
1. It does not allow for much review .
2. Poor choice for large scale projects
3. High risk and uncertainty
4. Can't test the software again and again

3.What is software engineering and why is it relevant ?

Software engineering is a process where you can build engineering design, develop and maintain software.
It analyses user needs and user application and design a software that will satisfy these needs.
It is important as it is the backbone of software.
It contains simple programming and it is used for big software applications
It is very important as it reduces complexity, minimize software cost, decreases development time and have effectiveness towards software.