1. pleas answer these questions with complete explanation.

i.can we make a default method static in java?

متد های استاتیک به هیچ وجه نمیتوانند اورراید و یا ارث بری شوند در صورتی که متدهای دیفالت برای ارث بری شدن طراحی شدن و درکنار آن قابلیت اورراید شدن نیز دارند(مانند متد های غیر استاتیک)بنابر این معنی و مفهوم این دو کلمه کلیدی به با هم ناسازگار است و غیر قابل انجام

ii.can an interface contain more than one default method?

بله.متد های دیفالت برای این به وجود آمده اند که بدون نیاز به پیاده سازی یک متد در چند کلاس مختلف که اینترفیس مورد نظر را ارث بری کرده اند بک متد را برای تمامی کلاس ها تعریف کنیم و در صورت نیاز آنرا اورراید کنیم

در کنار این منظور به ما این امکان را میدهد که برای برنامه های از قبل نوشته شدی متدی طراحی کنیم بدون نیاز به اینکه آنرا در کلاس ارث بری شده پیاده سازی کنیم.بنابر این توانایی ایجاد چند متد دیفالت در یک اینترفیس را داریم

iii.Can default method override equals(), hashCode() or toString() from Object class?

خیر.یک اینترفیس نمیتواند متد های دیفالت برای هیچ یک از متد های کلاس object پیاده سازی کند.این اتفاق نتیجه قانون "class wins" میباشد(یک متد که در"superclass" زنجیره قرار دارد همیشه نسبت به هر متدی در "superinterface" حق تقدم دارد  
) و این به این معنیست که کسی نمیتواند در داخل یک اینترفیس متد دیفالتی برای اورراید کردن متد های مانند equals(), hashCode() or toString پیاده سازی کند.

اما این این قانون در بعضی جاها نقض شده مانند کلاس List که متد equals در داخل کلاس پیاده سازی شده است.پس چرا این توانایی از دییگر اینترفیس ها گرفته شده است؟

دلیل اول:تمامی نمونه هایی که از یک اینترفیس ساته میشوند کلاس پایینی کلاس object میباشند و یک پیاده سازی غیر دیفالت از equals,hashcode,tostring دارند پس یک ورژن دیفالت داخل یک اینترفیس همیشه بیهودست و حتی احتمال کامپایل نشدن نیز هست

دلیل دوم:پیاده سازی یک متد دیفالت در اینترفیس از این متد ها(equals,tostring,hashcode) به احتمال زیاد اشتباه است.به دلیل اینکه این متد ها متدهایی هستند که محاسباتی بر اساس وضعیت یک شئ انجام میدهند و اینترفیس در اصل به همچنین چیزی دسترسی ندارد و فقط خود کلاسی که شئ را پیاده سازی میکند به وضعیت شئ دسترسی دارد و باید این متد هارا پیاده سازی کند.بنابر این احتمال اینکه متد دیفالتی برای چنین شرایط در یک اینترفیس استفاده شود خیلی کم است.

iv.Can the default method be abstract in Java?

متد های abstract دارای بدنه نیستند و باید در کلاس های ارث برده شده پیاده سازی شوند در صورتی که متد های دیفالت دارای بدنه هستند و در کنار آن توانایی اورراید شدن نیز دارند .با توجه به تعاریف و کاربرد این دو نوع متد به سادگی جواب این سوال "خیر " میباشد.