1. Create a payload class with required attributes.

@Data  
@AllArgsConstructor  
@NoArgsConstructor  
@Builder  
public class MobileResponse {  
 private List<MobileVo> content;  
 private int pageNo;  
 private int pageSize;  
 private long totalCount;  
 private int totalPage;  
 private boolean isLastPage;  
  
}

1. Define methods in Service interface with different parameters as per requirements

public interface MobileService {  
 List<MobileVo> findAllMobiles();  
 MobileResponse findAllMobiles(int pageNo, int pageSize);  
 MobileResponse findAllMobiles(int pageNo, int pageSize,

String sortBy, String sortOrder);

1. Implement the interface in service implementation class

@Override  
public MobileResponse findAllMobiles(int pageNo, int pageSize, String sortBy, String sortOrder) {  
 Pageable pageable=null;  
 if(sortBy!=null && sortOrder!=null) {  
 Sort sort = sortOrder.equalsIgnoreCase("asc") ? Sort.*by*(sortBy).ascending() : Sort.*by*(sortBy).descending();  
 pageable = PageRequest.*of*(pageNo, pageSize, sort);  
 }else{  
 pageable = PageRequest.*of*(pageNo, pageSize);  
 }  
 List<MobileVo> mobileVos = new ArrayList<>();  
 Page<MobileEntity> mobileEntities = mobileRepository.findAll(pageable);  
 if (mobileEntities != null) {  
 for (MobileEntity mobileEntity : mobileEntities) {  
 MobileVo mobileVo = new MobileVo();  
 BeanUtils.*copyProperties*(mobileEntity, mobileVo);  
 mobileVos.add(mobileVo);  
 }  
 MobileResponse mobileResponse = MobileResponse.*builder*()  
 .content(mobileVos)  
 .pageNo(mobileEntities.getNumber())  
 .pageSize(mobileEntities.getSize())  
 .totalPage(mobileEntities.getTotalPages())  
 .totalCount(mobileEntities.getTotalElements())  
 .isLastPage(mobileEntities.isLast())  
 .build();  
 return mobileResponse;  
 }  
 return null;  
}

1. Create controller class

@RestController  
@RequestMapping("/api/v2")  
public class MobileControllerV2 {  
 @Autowired  
 private MobileService mobileService;  
  
  
 @GetMapping("/mobiles")  
 public MobileResponse findAllMobiles(@RequestParam(name = "pageNo", defaultValue = "0", required = false) int pageNo,  
 @RequestParam(name = "pageSize", defaultValue = "10", required = false) int pageSize,  
 @RequestParam(name = "sortBy", defaultValue = "vendor", required = false)String sortBy,  
 @RequestParam(name = "sortOrder", defaultValue = "asc", required = false)String sortOrder){  
 return mobileService.findAllMobiles(pageNo,pageSize, sortBy, sortOrder);  
 }  
 }