1. Write a single query to display all Categories, Sub-Categories & their Product names whose stock level is greater than 0 (Use joins, avoid using multiple query)

Select C.category, S.subCategory, I.name, I.issueDate, I.price, I.stock from items I

inner join subcategories S on I.subCatId = S.id

inner join categories C on S.categoryId = C.id

where I.stock > 0

1. Write a single query to display all the product details (with their relevant Category & Sub-categories) whose category id is 1 & 3. Also display the products whose status is 1 (Use joins or Sub-queries, avoid using multiple queries)

Select C.category, S.subCategory, I.name, I.issueDate, I.price, I.stock from items I

inner join subcategories S on I.subCatId = S.id

inner join categories C on S.categoryId = C.id

where C.id in (1,3) and I.status = 1

1. Write a query to display the list of products which are added in the Month of April & 2011.

Select C.category, S.subCategory, I.name, I.issueDate, I.price, I.stock, Year(I.issueDate), month(I.issueDate) from items I

inner join subcategories S on I.subCatId = S.id

inner join categories C on S.categoryId = C.id

where Year(I.issueDate) = 2011 and month(I.issueDate) = 4

1. Display the products details whose price value range is between 10 and 60 (Avoid using greater / lesser than operators, use BETWEEN in the MySQL Query)

Select C.category, S.subCategory, I.name, I.issueDate, I.price, I.stock, Year(I.issueDate), month(I.issueDate) from items I

inner join subcategories S on I.subCatId = S.id

inner join categories C on S.categoryId = C.id

where I.price between 10 and 60

1. Display the latest added product details along with their issue date (Use MySQL function for displaying the Issue date format as 19 January, 11).

Select C.category, S.subCategory, I.name, I.price, I.stock, DATE\_Format(I.issueDate, '%M %e, %y') as issueDate from items I

inner join subcategories S on I.subCatId = S.id

inner join categories C on S.categoryId = C.id

ORDER BY I.issueDate DESC