



Junior Developer Test Task

Welcome to Scandiweb Junior Developer test assignment!

Thank you for your interest and time! Since 2003 we have been figuring out what is the minimum skills and knowledge level to be successful at your training and further career at scandiweb.

This test requires showcasing this minimum skill level by completing practical tasks. It helps you to check whether your level is already good to join us as a Junior Developer.

Expected outcome of the test

A web-app containing two pages for

1. Product list page
2. Adding a product page

Product list

The screenshot shows a "Product List" interface with a header bar containing back, forward, search, and refresh icons, along with a URL bar showing <https://juniorname.surname.com/product/list>. Below the header is a toolbar with "ADD" and "MASS DELETE" buttons. The main area displays a grid of products in three rows:

- Row 1:** Four items, each with a checkbox and a product ID followed by "Acme DISC", price "1.00 \$", and size "Size: 700 MB".
- Row 2:** Four items, each with a checkbox and a product ID followed by "War and Peace", price "20.00 \$", and weight "Weight: 2KG".
- Row 3:** Four items, each with a checkbox and a product ID followed by "Chair", price "40.00 \$", and dimension "Dimension: 24x45x15".

At the bottom of the list area, there is a footer note: "Scandiweb Test assignment".

[Open full size image](#)

The list items should contain the following details

- SKU (unique for each product)
- Name
- Price in \$
- One of the product-specific attributes and its value
 1. Size (in MB) for DVD-disc
 2. Weight (in Kg) for Book
 3. Dimensions (HxWxL) for Furniture

UI elements should be

- “ADD” button, which would lead to the “Product Add” page
- “MASS DELETE” action, implemented as checkboxes next to each product and a button triggering delete action for the selected products.
- There should be no pagination for the listing, all items should be on the same page
- Do not show any notification messages or alert windows loading the list or after submitting ADD new product dialogue

Adding a product page

Product Add
<https://juniorstest.name.surname.com/product/add>

Product Add

Save **Cancel**

SKU	<input type="text"/>
Name	<input type="text"/>
Price (\$)	<input type="text"/>
Type Switcher	Type Switcher ▾
<p>The form should be dynamically changed when type is switched</p> <div style="background-color: #ccc; height: 100px; width: 100%;"></div> <p>Special attribute should have a description with helpful information, related to its type, e.g.: "Please provide dimensions in HxWxL format", when type: Furniture is selected</p>	
<div style="border: 1px solid black; padding: 5px;"> <p>Size (MB) <input type="text"/></p> <p>"Product description"</p> </div> <div style="border: 1px solid black; padding: 5px;"> <p>Height (CM) <input type="text"/></p> <p>Width (CM) <input type="text"/></p> <p>Length (CM) <input type="text"/></p> <p>"Product description"</p> </div> <div style="border: 1px solid black; padding: 5px;"> <p>Weight (KG) <input type="text"/></p> <p>"Product description"</p> </div>	

Scandiweb Test assignment

[Open full size image](#)

The page should display a form, with the following fields

- SKU

- Name
- Price
- Product type switcher (DVD / Book / Furniture)
- Product type-specific attribute:
 - Size (in MB) for DVD-disc
 - Weight (in Kg) for Book
 - Dimensions (HxWxL) for Furniture

Add product page requirements

- The form should be dynamically changed when type is switched
- Special attributes should have a description, related to its type, e.g.: “Please, provide dimensions” / “Please, provide weight” / “Please, provide size” when related product type is selected
- All fields are mandatory for submission, missing values should trigger notification “Please, submit required data”
- Implement input field value validation, invalid data must trigger notification “Please, provide the data of indicated type”
- Notification messages should appear on the same page without reload
- The page must have a “Save” button to save the product. Once saved, returned to the “Product List” page with the new product added.
- The page must have a “Cancel” button to cancel adding the product action. Once cancelled, returned to the “Product List” page with no new products added.
- No additional dialogues like “Are you sure you want to Save / Cancel?”

Requirements for the coding practice

- Utilize OOP principles to handle differences in type logic / behaviour
- Meet PSR standards (<https://www.php-fig.org>)
- Avoid using conditional statements for handling differences in product types

- PHP: ^7.0, plain classes, no frameworks, OOP approach
- jQuery: optional
- jQuery-UI: prohibited
- Bootstrap: optional
- SASS: advantage
- MySQL: ^5.6 obligatory

How to submit?

1. Send the URL, where the above two pages “Product List” and “Add Product” will be available without password and ready to be used.

Make sure the URL with this app is available 24/7 and is not dependent on your computer being “on” and connected. Only external deployment.

You can use your preferred or this free PHP and MySQL hosting:

<https://www.000webhost.com/>.

2. Share the code as Bitbucket repository that is shared with a user with hr@scandiweb.com email.

Good Luck!



Well, we just looked up some random image in Unsplash for "good luck" and here it is, the Chinese dragon!



and that's our HR team celebration of a new hire

