**Question 1**

Assume that your app contains a RecyclerView that displays items in a shopping list. Your app also defines a click-listener class:

class ShoppingListItemListener(val clickListener: (itemId: Long) -> Unit) {

fun onClick(cartItem: CartItem) = clickListener(cartItem.itemId)

}

How do you make the ShoppingListItemListener available to data binding? Select one.

▢ In the layout file that contains the RecyclerView that displays the shopping list, add a <data> variable for ShoppingListItemListener.

▢🡺 In the layout file that defines the layout for a single row in the shopping list, add a <data> variable for ShoppingListItemListener.

▢ In the ShoppingListItemListener class, add a function to enable data binding:

fun onBinding (cartItem: CartItem) {dataBindingEnable(true)}

▢ In the ShoppingListItemListener class, inside the onClick() function, add a call to enable data binding:

fun onClick(cartItem: CartItem) = {

clickListener(cartItem.itemId)

dataBindingEnable(true)

}

**Question 2**

Where do you add the android:onClick attribute to make items in a RecyclerView respond to clicks? Select all that apply.

▢ In the layout file that displays the RecyclerView, add it to <androidx.recyclerview.widget.RecyclerView>

▢🡺 Add it to the layout file for an item in the row. If you want the entire item to be clickable, add it to the parent view that contains the items in the row.

▢🡺 Add it to the layout file for an item in the row. If you want a single TextView in the item to be clickable, add it to the <TextView>.

▢ Always add it the layout file for the MainActivity.