

Pratica 11

1) a) O botão add_item adiciona o item a base de dados do programa, enquanto o fechar pedido finaliza o pedido somando o preço total.

(outras questões no código final)

Código final:

dynamic item/3.

dynamic pedido/2.

dynamic telefone/2.

dynamic procurar/2.

controle_pedidos :-

```
new(D, dialog('Registra')),
send(D, append(new(Nome, text_item(nome)))),
send(D, append(new(Produto, text_item(produto)))),
send(D, append(new(Qtde, text_item(qtde)))),
send(D, append(new(Preco, text_item(preco)))),
send(D, append(button(add_item, and(message(@prolog, assert_item,
    Produto?selection,
    Qtde?selection,
    Preco?selection),
    message(Produto, clear),
    message(Qtde, clear),
    message(Preco, clear),
    message(Produto, keyboard_focus, true))))),
send(D, append(button(fechaPedido, and(message(@prolog, assert_pedido, Nome?selection),
    message(Nome, clear),
    message(Nome, keyboard_focus, true),
    message(@prolog, totaliza, Nome?selection))))),
```

```

    send(D, append(button(cancel, message(D, destroy)))),
    send(D, default_button(add_item)),
    send(D, open),
    retractall(pedido(_, _)),
    retractall(item(_, _)).

```

assert_item(Prod, Qtd, Preço) :-

```

    atom_number(Qtd, Q), atom_number(Preço, P),
    assert(item(Prod, Q, P)).

```

assert_pedido(Nome) :-

```

    findall(item(A,B,C), item(A,B,C), Itens),
    assert(pedido(Nome, Itens)).

```

totaliza(Nome) :-

```

    totaliza_pedido(Nome, Total),
    new(J, frame('Totalizacao')),
    new(D, dialog),
    atom_number(T, Total),
    send(D, append(text_item('Total do Pedido: ', T))),
    send(J, append, D),
    send(J, open).

```

totaliza_pedido(Pessoa, Total) :-

```

    pedido(Pessoa, Listaltens),
    totaliza_lista(Listaltens, Total).

```

totaliza_lista([], 0).

totaliza_lista([item(_, Qtd, Preço) | R], Total) :-

```

    totaliza_lista(R, TotalParcial),
    Total is TotalParcial + Qtd*Preço.

```

assert_telefone(Nome, Telefone) :-

assert(telefone(Nome, Telefone)).

procura_pessoa(Nome):-

procurar(Nome,_).

procurar(Nome, Telefone) :-

telefone(Nome,X),

Telefone = X,

new(J, frame("janela de telefone")),

new(C, dialog),

send(C, append(text_item('Telefone', Telefone))),

send(J, append, C),

send(J, open).

agenda_contatos :-

new(T, dialog('Registra')),

send(T, append(new(Nome, text_item(nome)))),

send(T, append(new(Telefone, text_item(telefone)))),

send(T, append(button(add_pessoa, and(message(@prolog, assert_telefone,

Nome?selection,

Telefone?selection),

message(Nome, clear),

message(Telefone, clear),

message(Nome, keyboard_focus, true))))),

send(T, append(button(cancel, message(T, destroy)))),

send(T, default_button(add_pessoa)),

send(T,open).

procurar_contato :-

new(Y, dialog('Procurar')),

```
send(Y, append(new(Nome, text_item(nome)))),  
send(Y, append(button(procurar, and(message(@prolog, procura_pessoa,  
Nome?selection))))),  
send(Y, append(button(cancel, message(Y, destroy)))),  
send(Y, default_button(procurar)),  
send(Y, open).
```