```cpp
void Application::chair(float seat_width, float seat_height, float seat_depth,
    float seat_x_position, float seat_y_position, float seat_z_position,
    float seat_x_rotation, float seat_y_rotation, float seat_z_rotation,
    float seat_x_rotation_2, float seat_y_rotation_2, float seat_z_rotation_2,
    float leg_width_depth, float leg_height,
    float b_support_height, float b_support_rest_height, float chair_uniform_scale) {

    // draw seat
    bm_chair_object_ptr = new basic_model;  // declare basic_model pointer to allocate on the heap

    bm_chair_object_ptr->mesh = rectangular_prism(            // create the mesh
        seat_width, seat_height, seat_depth,
        seat_x_position, seat_y_position, seat_z_position,
        seat_x_rotation, seat_y_rotation, seat_z_rotation,
        seat_x_rotation_2, seat_y_rotation_2, seat_z_rotation_2,
        chair_uniform_scale
    );
    bm_chair_object_ptr->color = vec3(0.55, 0.05, 0.05);    // assign color to object
    bm_chair_object_ptr->shader = just_shader;    // assign shader to object

    // bm_object_ptr is a basic_model*; declared application.hpp
    // initializes current basic_model pointer with the heap allocated basic_model object that is initialized with the current gl::mesh object, called mesh_object

    // m_all_objects is a vector<basic_model*>; declared application.hpp
    // stores all basic_model pointers to heap allocated basic_model objects
    m_chair_objects.push_back(bm_chair_object_ptr);      // append current basic_model object

    // draw 4 legs--------------------------------
    // leg 1 (+,-,+)--------------------------------
    bm_chair_object_ptr = new basic_model;
    bm_chair_object_ptr->mesh = rectangular_prism(leg_width_depth, leg_height, leg_width_depth,     // create the mesh
        seat_x_position + (seat_width * 0.5) - (leg_width_depth * 0.5),
        seat_y_position - (leg_height * 0.5) - (seat_height * 0.5),
        seat_z_position + (seat_depth * 0.5) - (leg_width_depth * 0.5),
        seat_x_rotation, seat_y_rotation, seat_z_rotation,
        seat_x_rotation_2, seat_y_rotation_2, seat_z_rotation_2, chair_uniform_scale);
    bm_chair_object_ptr->color = vec3(0.05, 0.05, 0.05);    // assign color to object
    bm_chair_object_ptr->shader = just_shader;    // assign shader to object
    m_chair_objects.push_back(bm_chair_object_ptr);      // append current basic_model object

    // leg 2 (-,-,+)--------------------------------
    bm_chair_object_ptr = new basic_model;
    bm_chair_object_ptr->mesh = rectangular_prism(leg_width_depth, leg_height, leg_width_depth,     // create the mesh
        seat_x_position - (seat_width * 0.5) + (leg_width_depth * 0.5),
        seat_y_position - (leg_height * 0.5) - (seat_height * 0.5),
        seat_z_position + (seat_depth * 0.5) - (leg_width_depth * 0.5),
        seat_x_rotation, seat_y_rotation, seat_z_rotation,
        seat_x_rotation_2, seat_y_rotation_2, seat_z_rotation_2, chair_uniform_scale);

    bm_chair_object_ptr->color = vec3(0.05, 0.05, 0.05);    // assign color to object
    bm_chair_object_ptr->shader = just_shader;    // assign shader to object
    m_chair_objects.push_back(bm_chair_object_ptr);       // append current basic_model object

    // leg 3 (-,-,-)--------------------------------
    bm_chair_object_ptr = new basic_model;
    bm_chair_object_ptr->mesh = rectangular_prism(leg_width_depth, leg_height, leg_width_depth,      // create the mesh
        seat_x_position - (seat_width * 0.5) + (leg_width_depth * 0.5),
        seat_y_position - (leg_height * 0.5) - (seat_height * 0.5),
        seat_z_position - (seat_depth * 0.5) + (leg_width_depth * 0.5),
        seat_x_rotation, seat_y_rotation, seat_z_rotation,
        seat_x_rotation_2, seat_y_rotation_2, seat_z_rotation_2, chair_uniform_scale);

    bm_chair_object_ptr->color = vec3(0.05, 0.05, 0.05);    // assign color to object
    bm_chair_object_ptr->shader = just_shader;    // assign shader to object
    m_chair_objects.push_back(bm_chair_object_ptr);      // append current basic_model object

    // leg 4 (+,-,-)--------------------------------
    bm_chair_object_ptr = new basic_model;
    bm_chair_object_ptr->mesh = rectangular_prism(leg_width_depth, leg_height, leg_width_depth,      // create the mesh
        seat_x_position + (seat_width * 0.5) - (leg_width_depth * 0.5),
        seat_y_position - (leg_height * 0.5) - (seat_height * 0.5),
        seat_z_position - (seat_depth * 0.5) + (leg_width_depth * 0.5),
        seat_x_rotation, seat_y_rotation, seat_z_rotation,
        seat_x_rotation_2, seat_y_rotation_2, seat_z_rotation_2, chair_uniform_scale);

    bm_chair_object_ptr->color = vec3(0.05, 0.05, 0.05);    // assign color to object
    bm_chair_object_ptr->shader = just_shader;    // assign shader to object
    m_chair_objects.push_back(bm_chair_object_ptr);       // append current basic_model object
    //-----------------------------------------------

    // draw back support beams
    // leg 5 (+,-,-)--------------------------------
    bm_chair_object_ptr = new basic_model;
    bm_chair_object_ptr->mesh = rectangular_prism(leg_width_depth, b_support_height, leg_width_depth,    // create the mesh
        seat_x_position + (seat_width * 0.5) - (leg_width_depth * 0.5),
        seat_y_position + (b_support_height * 0.5) + (seat_height * 0.5),
        seat_z_position - (seat_depth * 0.5) + (leg_width_depth * 0.5),
        seat_x_rotation, seat_y_rotation, seat_z_rotation,
        seat_x_rotation_2, seat_y_rotation_2, seat_z_rotation_2, chair_uniform_scale);

    bm_chair_object_ptr->color = vec3(0.05, 0.05, 0.05);    // assign color to object
    bm_chair_object_ptr->shader = just_shader;    // assign shader to object
    m_chair_objects.push_back(bm_chair_object_ptr);       // append current basic_model object

    // leg 6 (-,-,-)--------------------------------
    bm_chair_object_ptr = new basic_model;
    bm_chair_object_ptr->mesh = rectangular_prism(leg_width_depth, b_support_height, leg_width_depth,     // create the mesh
        seat_x_position - (seat_width * 0.5) + (leg_width_depth * 0.5),
        seat_y_position + (b_support_height * 0.5) + (seat_height * 0.5),
        seat_z_position - (seat_depth * 0.5) + (leg_width_depth * 0.5),
        seat_x_rotation, seat_y_rotation, seat_z_rotation,
        seat_x_rotation_2, seat_y_rotation_2, seat_z_rotation_2, chair_uniform_scale);

    bm_chair_object_ptr->color = vec3(0.05, 0.05, 0.05);    // assign color to object
    bm_chair_object_ptr->shader = just_shader;    // assign shader to object
    m_chair_objects.push_back(bm_chair_object_ptr);       // append current basic_model object

    // back rest 7 (0,-,-)--------------------------------
    bm_chair_object_ptr = new basic_model;
    bm_chair_object_ptr->mesh = rectangular_prism(seat_width - leg_width_depth, b_support_rest_height, leg_width_depth * 0.7,     // create the mesh
        seat_x_position,
        seat_y_position + (b_support_height * 0.5) + (seat_height * 0.5),
        seat_z_position - (seat_depth * 0.5) + (leg_width_depth * 0.5),
        seat_x_rotation, seat_y_rotation, seat_z_rotation,
        seat_x_rotation_2, seat_y_rotation_2, seat_z_rotation_2, chair_uniform_scale);

    bm_chair_object_ptr->color = vec3(0.55, 0.05, 0.05);    // assign color to object
    bm_chair_object_ptr->shader = just_shader;    // assign shader to object
    m_chair_objects.push_back(bm_chair_object_ptr);       // append current basic_model object


}
```