

```
/**
 * The <code>registerPublishFloatAlert</code> method is to register a publish with the edgeStatusRouter.
 * <BR>
 * @param variableName The name of the variable which event is to be published.
 * @param minInterval The minimum interval between two alerts being published. This will
 * prevent the publisher from flooding the network.
 * @return Returns the variableId of the registered publication.
 * @throws PublisherException If something went wrong during the registration.
 */
public int registerPublishFloatAlert(String variableName, int minInterval) throws PublisherException;

/**
 * The <code>registerPublishBooleanAlert</code> method is to register a publish with the edgeStatusRouter.
 * <BR>
 * @param variableName The name of the variable which event is to be published.
 * @param minInterval The minimum interval between two alerts being published. This will
 * prevent the publisher from flooding the network.
 * @return Returns the variableId of the registered publication.
 * @throws PublisherException If something went wrong during the registration.
 */
public int registerPublishBooleanAlert(String variableName, int minInterval) throws PublisherException;

/**
 * The <code>registerPublishUserDefinedAlert</code> method is to register a publish with the edgeStatusRouter.
 * <BR>
 * @param variableName The name of the variable which event is to be published.
 * @param minInterval The minimum interval between two alerts being published. This will
 * prevent the publisher from flooding the network.
 * @param userType The type of the user registered type.
 * @param length The length in bytes of the user defined type.
 * @return Returns the variableId of the registered publication.
 * @throws PublisherException If something went wrong during the registration.
 */
public int registerPublishUserDefinedAlert(String variableName, int minInterval, int userType, int length) throws PublisherException;

/**
 * The <code>unregisterPublish</code> method is to unregister a previous publish with the edgeStatusRouter.
 * <BR>
 * @param variableId The id of the message to be published.
 * @return Returns <code>0</code> if unregistered from ControlServer,
 * error code otherwise.
 */
public short unregisterPublish(int variableId);
}
```