static BlockingCollection<string> queue = new BlockingCollection<string>();

static ExchangeService service;

StartExchangeService();

GetEmails();

var folder = Folder.Bind(service, WellKnownFolderName.DeletedItems);

foreach (var item in queue.GetConsumingEnumerable())

{

    ProcessEmail(item);

    Item item2 = Item.Bind(service, item.ToString());

    item2.Move(folder.Id);

}

static private void StartExchangeService()

  {

    try

      {

        service = new ExchangeService(ExchangeVersion.Exchange2010\_SP2);

        Uri uri = new Uri("<https://outlook.office365.com/EWS/Exchange.asmx>");

        IWebProxy proxy = WebRequest.GetSystemWebProxy();

        bool isBypassed = proxy.IsBypassed(uri);

        if (!isBypassed)

           {

              //if proxy is needed, then provide network credentials

              proxy.Credentials = CredentialCache.DefaultNetworkCredentials;

              service.WebProxy = proxy;

           }

         //https://msdn.microsoft.com/en-us/library/office/dn567668(v=exchg.150).aspx

#if DEBUG

    service.Credentials = new WebCredentials("[tdal6392@sa.convergys.com](mailto:tdal6392@sa.convergys.com)", "pvnptczcvqqwmlxy", "sa.convergys.com"); #else

    service.Credentials = new WebCredentials("[notification.robot@na.convergys.com](mailto:notification.robot@na.convergys.com)", "wjkxyfrdxgqpzxjy", "na.convergys.com");

#endif

         service.Url = uri;

        //AutodiscoverUrl doesn't work with O365

    }

    catch (Exception ex)

    {

         string msgErr = "StartExchangeService [\\n\\n](file:///\\n\n) " + ex.ToString();

         ReportError(msgErr);

    }

}

static private void GetEmails()

  {

      log.Info("Getting pending emails.");

      try

          {

             bool process = true;

             if (process)

             {

               int offset = 0;

               int pageSize = 50;

               bool more = false;

               ItemView view = new ItemView(pageSize, offset, OffsetBasePoint.Beginning);

               view.OrderBy.Add(ItemSchema.DateTimeReceived, SortDirection.Descending);

               view.PropertySet = PropertySet.IdOnly;

               FindItemsResults<Item> results;

               do

                {

                   results = service.FindItems(WellKnownFolderName.Inbox, view);

                   foreach (var item in results.Items)

                        {

                            queue.Add(item.Id.UniqueId);

                        }

                   more = results.MoreAvailable;

                   if (more)

                        {

                            view.Offset += pageSize;

                        }

                } while (more);

           }

      }

    catch (Exception ex)

      {

        string msgErr = "GetPendingEmails\\n\\n " + ex.ToString();

      }

  }

static private void ProcessEmail(string itemId)

        {

            Item item = null;

            StringBuilder sb = new StringBuilder();

            try

            {

                item = Item.Bind(service, itemId);

                if (item is EmailMessage)

                {

                    EmailMessage email = item as EmailMessage;

                    if (string.IsNullOrEmpty(email.Subject))

                    {

                        DataAccess.SendEmail("Error EmplVerif", "Missing subject", "");

                        return;

                    }

                    Email e = new Email();

                    string sender = email.Sender.Address;

                    string sendername = email.Sender.Name;

                    string dtSended = email.DateTimeSent.ToString().Trim();

                    string subject = email.Subject.ToUpper();

                    string body = email.Body.Text;

                    string emailCreated = email.DateTimeCreated.ToString();

                    int largo = body.Length;

                    HtmlAgilityPack.HtmlDocument doc = new HtmlAgilityPack.HtmlDocument();

                    doc.LoadHtml(body);

                    string msg1 = doc.DocumentNode.SelectSingleNode("//body").InnerText;

                    msg1 = Regex.Replace(msg1, @"\s+", " ").Trim();

                    log.Info(sender + " " + subject);

                    File1 = e.Download(email, "");

                    if (File1 == "")

                    {

                        File1 = e.Download2(email);

                        if (File1 == "")

                        {

                            DataAccess.SendEmail("Missing Attachement","There is not a document attached", sender);

                            return;

                        }

                    }

                    File2 = e.Download(email, File1);

                    if (File2 == "") File2 = e.Download2(email);

                    if (File2 == File1) File2 = "";

                    int pos = msg1.IndexOf("Employee Name");

                    if (pos < 1) return;

                    msg1 = msg1.Substring(pos);

                    pos = body.IndexOf("Middle Name");

                    string names1 = body.Substring(pos);

                    pos = names1.IndexOf("Date of Birth");

                    names1 = names1.Substring(0, pos);

                    names1 = names1.Replace("\r\n", "");

                    names1 = names1.Replace("\*", "");

                    pos = names1.IndexOf("<td");

                    names1 = names1.Substring(pos);

                    string last = GetName(names1);

Chrome to connect to outlook

Go to <https://outlook.office.com/owa/>

Log in as [notification.robot@na.convergys.com](mailto:notification.robot@na.convergys.com)

Password is n0tifr0b0t!

service.Credentials = new WebCredentials("[notification.robot@na.convergys.com](mailto:notification.robot@na.convergys.com)", "wjkxyfrdxgqpzxjy", "na.convergys.com");

1. Go to <https://portal.office.com/account/>

2. Go to Security & privacy

3. Go to Additional security verification

4. Go to Create and manage app passwords

   service.Credentials = new WebCredentials("[tdal6392@sa.convergys.com](mailto:tdal6392@sa.convergys.com)", "pvnptczcvqqwmlxy", "sa.convergys.com"); //enter your credentials to test