

UQP - Microsoft Visual Studio

File Edit View Project Build Debug Team Tools Test Analyze Window Help

Debug Any CPU IS Express (Google Chrome) 192.61.99.225

MRAController.cs MembersController.cs Source Control Explorer ACO\_MembersMart.cs Notification\_UserSpecific.cs

UQP UQP Controllers.MRAController Healthplan(HealthplanMRA healthplan)

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Net;
5  using System.Net.Http;
6  using System.Web;
7  using System.Web.Http;
8  using UQP.Models;
9  using UQP.Utility;
10 using System.Linq.Dynamic;
11 //using System.Linq.Dynamic.Core;
12 using System.Data.Entity.Core.Objects;
13
14
15 namespace UQP.Controllers
16 {
17     [Authorize]
18     [References]
19     public class MRAController : ApiController
20     {
21         //private UQPEntities db = new UQPEntities(HttpContext.Current.Request.Headers.GetValues("IPA").FirstOrDefault());
22         //private UQPEntities db = new UQPEntities();
23         [HttpPost]
24         [Route("api/mra/HealthplanScorecard")]
25         [References]
26         public HttpResponseMessage Healthplan(HealthplanMRA healthplan)
27         {
28             try
29             {
30                 using (UQPEntities db = new UQPEntities(HttpContext.Current.Request.Headers.GetValues("IPA").FirstOrDefault()))
31                 {
32                     HealthplanMRA healthplandetails = new HealthplanMRA();
33                     dynamic query = null;
34                     var lastmonth = DateTime.Now.AddMonths(-1).Month;
35                     if (healthplan.providerid == 0)
36                     {
37                         query = db.Healthplans
38                             .Where("it.ProviderID == " + providerid)
39                             .Where("it.Month == " + lastmonth)
40                             .Select("it");
41                         healthplandetails = query.FirstOrDefault();
42                     }
43                     else
44                     {
45                         query = db.Healthplans
46                             .Where("it.ProviderID == " + providerid)
47                             .Where("it.Month == " + lastmonth)
48                             .Select("it");
49                         healthplandetails = query.FirstOrDefault();
50                     }
51                     return Request.CreateResponse(HttpStatusCode.OK, healthplandetails);
52                 }
53             }
54         }
55     }
56 }
```

Output Show output from: Source Control - Team Foundation

MRAController.cs + X MembersController.cs Source Control Explorer ACO\_MemberMait.cs Notification\_UserSpecific.cs

UQP

```
16  {
17      [Authorize]
18      [References]
19      public class MRAController : ApiController
20      {
21          //private UQPEntities db = new UQPEntities(HttpContext.Current.Request.Headers.GetValues("IPA").FirstOrDefault());
22          //private UQPEntities db = new UQPEntities();
23          [HttpPost]
24          [Route("api/mra/HealthplanScorecard")]
25          [References]
26          public HttpResponseMessage Healthplan(HealthplanMRA healthplan)
27          {
28              try
29              {
30                  using (UQPEntities db = new UQPEntities(HttpContext.Current.Request.Headers.GetValues("IPA").FirstOrDefault()))
31                  {
32                      HealthplanMRA healthplandetails = new HealthplanMRA();
33                      dynamic query = null;
34                      var lastmonth = DateTime.Now.AddMonths(-1).Month;
35                      if (healthplan.providerid == 0)
36                      {
37
38                          query = (from h in db.MRA_ReportByInsurance.AsNoTracking()
39                                    join dash in db.NewDashboardsSuperset on
40                                    h.HealthplanId equals dash.Healthplan_Id
41                                    where h.HealthplanId == healthplan.HealthplanId && h.Projectedscore_Year == healthplan.year
42                                    && dash.CalculatedMonth.Value.Year == DateTime.Now.Year
43                                    && dash.CalculatedMonth.Value.Month == lastmonth
44
45                          select new
46                          {
47                              h.HealthplanId,
48                              h.HealthplanName,
49                              h.CurrentScore,
50                              h.DemightedScore
51                          });
52
53                      }
54
55                  }
56
57              }
58
59          }
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100 %
```



Microsoft Visual Studio 192.61.89.225

File Edit View Project Build Debug Team Tools Test Analyze Window Help

UQP.cs MRAController.cs MembersController.cs Source Control Explorer ACO\_MembersMart.cs Notification\_UserSpecific.cs

UQP.Controllers.MRAController

```
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110

    h.HealthplanId,
    h.HealthplanName,
    h.CurrentScore,
    h.Projectedscore,
    Projectedscore_MissingConditions = h.Projectedscore_MissingMembercount,
    TotalMembership = pro.totalMembership,
    CurrentScore_MissingConditions = h.CurrentScore_MissingMembercount
}
).FirstOrDefault();

healthplandetails.HealthplanId = query.HealthplanId;
healthplandetails.HealthplanName = query.HealthplanName;
healthplandetails.CurrentScore = query.CurrentScore;
healthplandetails.Projectedscore = query.Projectedscore;
healthplandetails.CurrentMissingConditions = query.CurrentScore_MissingConditions;
healthplandetails.TotalMember = (query.TotalMembership == null) ? 0 : Convert.ToInt32(query.TotalMembership);
healthplandetails.Projectedscore_MissingConditions = query.Projectedscore_MissingConditions;

return Request.CreateResponse(HttpStatusCode.OK, healthplandetails);
}

catch (Exception ex)
{
    return Request.CreateResponse(HttpStatusCode.InternalServerError, ex.InnerException);
}
```

A screenshot of a Microsoft Visual Studio IDE interface. The title bar shows "Debug Any CPU". The tabs at the top include "controller.cs", "I.MembersController.cs", "Source Control Explorer", "ACO\_MembersMart.cs", "Notification\_UserSpecific.cs", and "UQP.Controllers.MRAController". The status bar at the bottom right says "Scanned with CamScanner".

```
10 [HttpGet]
11 [Route("api/mra/hccCategory/{hccCode}")]
12 References
13 public HttpResponseMessage GethccCategorys(string hccCode)
14 {
15     try
16     {
17         using (UQPEntities db = new UQPEntities(HttpContext.Current.Request.Headers.GetValues("IPA").FirstOrDefault()))
18         {
19
20             //var hcc = (from categ in db.MRADemographics.AsNoTracking()
21             //           where( (hccCode=="null" && categ.Variable.Contains("hcc")) || (hccCode != "null" && categ.Variable=="hcc"+hccCode))
22             //           select new { categ.Variable, categ.Description_Label }).OrderBy(e => e.Description_Label).ToList();
23
24
25
26
27             dynamic abc;
28             if (hccCode == "null")
29             {
30                 abc = (from categ in db.MRADemographics.AsNoTracking()
31                         where
32                             categ.Variable.Contains("hcc")
33                         select new { categ.Variable, categ.Description_Label }).OrderBy(e => e.Description_Label).ToList();
34             }
35             else
36             {
37                 if (hccCode.Contains("."))
38                 {
39                     hccCode = hccCode.Replace(".", "");
40                 }
41                 abc = (from categ in db.MRADiagnosisCodes.AsNoTracking()
42                         where
43                             categ.DiagnosisCode == hccCode
44
45             }
```

UQP - Microsoft Visual Studio

192.61.99.225

Edit View Project Build Debug Team Tools Test Analyze Window Help

Debug Any CPU IIS Express (Google Chrome)

MRAController.cs MembersController.cs Source Control Explorer ACO\_MembersMart.cs Notification\_UserSpecific.cs

UQP UQP.Controllers.MRAController Healthplan(HealthplanMRA healthplan)

```
143     abc = (from categ in db.MRADiagnosisCodes.AsNoTracking()
144             where
145                 categ.DiagnosisCode == hccCode
146             select new { Variable = categ.CMSHCCModelCategoryV22, Description_Label = categ.Description }).ToList();
147     }
148
149
150
151     List<hccCategories> hcclist = new List<hccCategories>();
152     foreach (var per in abc)
153     {
154         hccCategories hccobj = new hccCategories
155         {
156             hccCategory = per.Variable,
157             Description = per.Description_Label
158         };
159         hcclist.Add(hccobj);
160     }
161
162     return Request.CreateResponse(HttpStatusCode.OK, hcclist);
163 }
164 }
165 catch (Exception ex)
166 {
167     return Request.CreateResponse(HttpStatusCode.BadRequest, ex);
168 }
169 }
170 }
171 }
172 }
```

Microsoft Visual Studio

File View Project Build Debug Team Tools Test Window Help

Debug Any CPU IIS Express (Google Chrome)

MRAController.cs MembersController.cs ACO\_MembersMart.cs Notification\_UserSpecific.cs

UQP

```
173 [HttpGet]
174 [Route("api/mra/members/{reporttype}")]
175 public HttpResponseMessage GetMemberDetails(string reporttype, [FromUri]HealthplanMRA member, [FromUri]PagingParameterModel pagingParameterModel)
176 {
177     try
178     {
179         using (UQPEntities db = new UQPEntities(HttpContext.Current.Request.Headers.GetValues("IPA").FirstOrDefault()))
180         {
181             db.Database.CommandTimeout = 240;
182             int totalcount = 0;
183             if (reporttype == "projectedscore" || reporttype == "currentscore")
184             {
185                 int fromyear = member.year;
186
187                 if (reporttype == "currentscore")
188                 {
189                     fromyear = member.year - 1;
190                 }
191
192                 dynamic query = null;
193                 string toBeSearched = null;
194                 string hccoutput = null;
195
196                 if (member.hccccategory != null)
197                 {
198                     member.hccccategory = Convert.ToString(member.hccccategory);
199                     toBeSearched = "HCC";
200                     hccoutput = member.hccccategory.Substring(member.hccccategory.IndexOf(toBeSearched) + toBeSearched.Length);
201                     var x = member.hccccategory.IndexOf(toBeSearched);
202                     var y = toBeSearched.Length;
203                 }
204
205                 List<MRAmembers> memberlist = new List<MRAmembers>();
206             }
207         }
208     }
209 }
```

Search Solution



```
229
230
231     member.gender, member.mintage, member.maxage, member.dob, member.filtertype,
232     pagingParameterModel.pageNumber, pagingParameterModel.pageSize, member.startdate,
233     member.enddate, totalmembercount).ToList();
234
235     totalCount = Convert.ToInt32(totalmembercount.Value);
236
237 } else
238 {
239     query = (from m in db.MRA_Members_FinalScore.AsNoTracking()
240     where m.isPresentInNM == true && (member.HealthplanId == 0 || m.HealthPlanID == member.HealthplanId)
241     && m.year == fromyear &&
242     (member.status == null || m.Status == member.status)
243     && (member.dob == null || m.DOB == member.dob)
244     && (member.providerid == 0 || m.provider_id == member.providerid)
245     && (member.gender == null || m.Gender == member.gender)
246     && (member.mintage == null || (m.Age >= member.mintage && m.Age <= member.maxage))
247     && (member.hccccategory == null || m.FinalHCCCcategory.StartsWith(hccoutput + ",") ||
248     m.FinalHCCCcategory == hccoutput ||
249     m.FinalHCCCcategory.Contains(", " + hccoutput + ",") ||
250     m.FinalHCCCcategory.EndsWith(", " + hccoutput))
251
252     && (member.hccnotfound == null || !(m.FinalHCCCcategory.StartsWith(member.hccnotfound + ",")) &&
253     m.FinalHCCCcategory != member.hccnotfound &&
254     !(m.FinalHCCCcategory.Contains(", " + member.hccnotfound + ",")) &&
255     !(m.FinalHCCCcategory.EndsWith(", " + hccoutput)))
256
257     && (member.membername == null || m.FirstName.StartsWith(member.membername) ||
258     m.LastName.StartsWith(member.membername) ||
259     (m.FirstName + " " + m.LastName).StartsWith(member.membername))
260     && (member.subsId == null || m.SubscriberId.StartsWith(member.subsId))
261     && (member.minscore == null || (m.FinalScore >= member.minscore && m.FinalScore < member.maxscore)
262 )
263 select new
264 {
265     SubscriberId = m.SubscriberId,
266     Age = m.Age,
267     Gender = m.Gender,
```

```
Microsoft Visual Studio
View Project Build Debug Team Tools Test Analyze Window Help
Debug Any CPU IIS Express (Google Chrome)
controller.cs* MembersController.cs Source Control Explorer ACO_MembersMart.cs Notification_UserSpecific.cs
UQP.Controllers.MRAController GetMemberDetails(string reporttype, Healthplan
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
    SubscriberId = m.SubscriberId,
    Age = m.Age,
    Gender = m.Gender,
    projectedscore = m.FinalScore,
    HealthPlanName = m.HealthPlanName,
    FirstName = m.FirstName,
    MidName = m.MidName,
    LastName = m.LastName,
    Enrollment_Status = m.Enrollment_Status,
    currentscore = m.FinalScore,
    demogscore = m.DemographicsScore,
    initialhcc = m.InitialHCCCcategory,
    droppedhcc = m.DroppedHCCCcategory,
    finalhcc = m.FinalHCCCcategory,
    finalcodes = m.FinalHCCCodes,
    scorebreakup = m.ClaimScoreBreakup,
    claimscore = m.ClaimsScore,
    provfirstname = m.ProviderFirstName,
    provmidname = m.ProviderMiddleName,
    provlastname = m.ProviderLastName,
    status = m.Status,
    dob = m.DOB

}).Distinct().OrderBy(e => new { e.LastName, e.FirstName }).ToList()
    .Skip(pagingParameterModel.pageSize * (pagingParameterModel.pageNumber - 1))
    .Take(pagingParameterModel.pageSize);
```

```
totalcount = (from m in db.MRA_Members_FinalScore.AsNoTracking()
    where m.IsPresentInMM == true && (member.HealthplanID == 0 || m.HealthPlanID == member.HealthplanID) &&
        m.year == fromyear && (member.status == null || m.Status == member.status)
        && (member.providerid == 0 || m.provider_id == member.providerid)
        && (member.dob == null || m.DOB == member.dob)
        && (member.gender == null || m.Gender == member.gender)
        && (member.mintage == null || (m.Age >= member.mintage && m.Age <= member.maxage))
        && (member.minscore == null || (m.FinalScore >= member.minscore && m.FinalScore < member.maxscore)) &&
            (member.membername == null || m.FirstName.StartsWith(member.membername) ||
            m.LastName.StartsWith(member.membername) ||
            (m.FirstName + " " + m.LastName).StartsWith(member.membername))
            && (member.subsId == null || m.SubscriberId.StartsWith(member.subsId)) &&
            (member.hcccategory == null || m.FinalHCCCategory.StartsWith(hccoutput + ",") ||
            m.FinalHCCCategory.Contains("," + hccoutput + ",") || m.FinalHCCCategory == hccoutput ||
            m.FinalHCCCategory.EndsWith(", " + hccoutput))
            && (member.hccnotfound == null || !(m.FinalHCCCategory.StartsWith(member.hccnotfound + ","))) &&
            m.FinalHCCCategory != member.hccnotfound &&
            !(m.FinalHCCCategory.Contains("," + member.hccnotfound + ",")) &&
            !(m.FinalHCCCategory.EndsWith("," + hccoutput)))
            select new
            {
                m.SubscriberId
            }).Distinct().Count();
        }
    }

else if (member.report == true)
{

```

Microsoft Visual Studio

File View Project Build Debug Team Tools Test Analyze Window Help

Debug Any CPU IIS Express (Google Chrome)

MRAController.cs MembersController.cs ACO\_MembersMart.cs Notification\_UserSpecific.cs

UQP

```
322
323     else if (member.report == true)
324     {
325         if ((reporttype == "currentscore" || reporttype == "projectedscore") && member.filtertype==null)
326         {
327
328             if (reporttype == "currentscore")
329             {
330                 query = db.USP_MRAMembersListing(member.membername, member.HealthplanId, member.subsId, fromyear, (fromyear + 1),
331                                                 member.minscore, member.maxscore, hccoutput, member.status, member.providerid,
332                                                 member.gender, member.mimage, member.maxage, member.dob).ToList();
333             }
334             else if (reporttype == "projectedscore")
335             {
336                 query = db.USP_MRAMembersListing(member.membername, member.HealthplanId, member.subsId, fromyear, (fromyear - 1),
337                                                 member.minscore, member.maxscore, hccoutput, member.status, member.providerid,
338                                                 member.gender, member.mimage, member.maxage, member.dob).ToList();
339             }
340             foreach (var q in query)
341             {
342                 MRAMembers memberdetail = new MRAMembers
343                 {
344                     SubscriberId = q.SubscriberId,
345                     Age = q.Age,
346                     Gender = q.Gender,
347                     HealthPlanName = q.HealthPlanName,
348                     FirstName = q.FirstName,
349                     //MidName = q.Midname,
350                     LastName = (q.LastName == null) ? "NA" : q.LastName,
351                     Enrollment_Status = q.Enrollment_Status,
352                     CurrentScore = (reporttype == "projectedscore") ? q.score2 : q.score1,
353                     ProjectedScore = (reporttype == "projectedscore") ? q.score1 : q.score2,
354                     DemographicsScore = q.DemographicsScore,
355                     InitialHCCcategory = q.InitialHCCcategory,
356                     DroppedHCCcategory = q.DroppedHCCcategory,
357                     FinalHCCcategory = q.FinalHCCcategory,
```

RAController.cs\* MembersController.cs Source Control Explorer ACO\_MembersMart.cs Notification\_UserSpecific.cs

```
355 InitialHCCcategory = q.InitialHCCcategory,
356 DroppedHCCcategory = q.DroppedHCCcategory,
357 FinalHCCcategory = q.FinalHCCcategory,
358 FinalHCCCodes = q.FinalHCCcodes,
359 description = (member.description == null) ? "NA" : member.description,
360 status = q.Status,
361 provfirstname = q.ProviderFirstName,
362 provlastname = (q.ProviderLastName == null) ? "NA" : q.ProviderLastName,
363 //provmidname = q.ProviderMiddleName,
364 dob = q.DOB
365
366 memberList.Add(memberdetail);
367 }
368 }
369 else if (member.filtertype == "snp" || member.filtertype == "nohcc")
370 {
371 if (member.filtertype == "snp")
372 {
373 ObjectParameter totalmembercount = new ObjectParameter("totalmembercount", typeof(Int32));
374 query = db.USP_CustomMRAMemberslisting(member.membername, member.HealthplanId,
375 member.subsId, fromyear, member.minscore, member.maxscore, hccoutput,
376 member.status, member.providerid,
377 member.gender, member.minage, member.maxage, member.dob, member.filtertype,
378 0, 0, member.startdate, member.enddate, totalmembercount).ToList();
379 }
380 if (member.filtertype == "nohcc")
381 {
382 ObjectParameter totalmembercount = new ObjectParameter("totalmembercount", typeof(Int32));
383 query = db.USP_CustomMRAMembersListing(member.membername, member.HealthplanId,
384 member.subsId, fromyear, member.minscore, member.maxscore, null,
385 member.status, member.providerid,
386 member.gender, member.minage, member.maxage, member.dob, member.filtertype,
387 0, 0, member.startdate, member.enddate, totalmembercount).ToList();
388 }
389
390 foreach (var q in query)
```

```
UQP.Controllers.cs  MembersController.cs  Source Control Explorer  ACO_MembersMart.cs  Notification_UserSpecific.cs  GetMemberDetails(string reporttype, HealthplanMRA member)
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
        member.status, member.providerid,
        member.gender, member.mintage, member.maxage, member.dob, member.filtertype,
        0, 0, member.startdate, member.enddate, totalmembercount).ToList();
    }

    foreach (var q in query)
    {
        MRAMembers memberdetail = new MRAMembers
        {
            SubscriberId = q.SubscriberId,
            Age = q.Age,
            Gender = q.Gender,
            HealthPlanName = q.HealthPlanName,
            FirstName = q.FirstName,
            //MidName = q.MidName,
            LastName = (q.LastName == null) ? "NA" : q.LastName,
            Enrollment_Status = q.Enrollment_Status,
            CurrentScore = q.currentscore,
            ProjectedScore = q.projectedscore,
            DemographicsScore = q.demogscore,
            DroppedHCCCcategory = q.droppedhcc,
            InitialHCCCcategory = q.initialhcc,
            FinalHCCCcategory = q.finalhcc,
            FinalHCCCodes = q.finalcodes,
            CodeBreakup = q.scorebreakup,
            provfirstname = q.provfirstname,
            //provmidname = q.provmidname,
            provlasttname = (q.provlasttname == null) ? "NA" : q.provlasttname,
            status = q.status,
            dob = q.dob
        };

        memberList.Add(memberdetail);
    }
}
```

```
421
422
423
424     if (member.report == false)
425     {
426         foreach (var q in query)
427         {
428             MRAMembers memberdetail = new MRAMembers
429             {
430                 SubscriberId = q.SubscriberId,
431                 Age = q.Age,
432                 Gender = q.Gender,
433                 HealthPlanName = q.HealthPlanName,
434                 FirstName = q.FirstName,
435                 //MidName = q.MidName,
436                 LastName = (q.LastName == null) ? "NA" : q.LastName,
437                 Enrollment_Status = q.Enrollment_Status,
438                 CurrentScore = q.currentscore,
439                 ProjectedScore = q.projectedscore,
440                 DemographicsScore = q.demogscore,
441                 DroppedHCCcategory = q.droppedhcc,
442                 InitialHCCcategory = q.initialhcc,
443                 FinalHCCCcategory = q.finalhcc,
444                 FinalHCCCCodes = q.finalcodes,
445                 CodeBreakup = q.scorebreakup,
446                 provfirstname = q.provfirstname,
447                 //provmidname = q.provmidname,
448                 provlasttname = (q.provlasttname == null) ? "NA" : q.provlasttname,
449                 status = q.status,
450                 dob = q.dob
451
452             };
453             memberList.Add(memberdetail);
454         }
455     }
456 }
```

File Edit View Tools Debug Any CPU IIS Express (Google Chrome) Source Control Explorer ACO\_MembersMart.cs Notification\_UserSpecific.cs

UQP.cs UQP.Controllers.MRAController GetMemberDetails(string reportType, HealthplanMRA member, PagingParameterModel)

```
457
458     #region pagination
459
460     if (member.report == false)
461     {
462
463         // Get's No of Rows Count
464         int count = totalcount;
465         //lstmembercount.Count();
466         // int count = 0;
467
468
469         // Parameter is passed from Query string if it is null then it default Value will be pageNumber:1
470         int currentPage = pagingParameterModel.pageNumber;
471
472         // Parameter is passed from Query string if it is null then it default Value will be pageSize:20
473         int pageSize = pagingParameterModel.pageSize;
474
475         // Display TotalCount to Records to User
476         int totalCount = count;
477
478         // Calculating Totalpage by Dividing (No of Records / Pagesize)
479         int totalPages = (int)Math.Ceiling(count / (double)pageSize);
480
481         // Returns List of Customer after applying Paging
482         var items = memberList;//.Skip((currentPage - 1) * pageSize).Take(pageSize).ToList();
483
484         // if currentPage is greater than 1 means it has previousPage
485         var previousPage = currentPage > 1 ? "Yes" : "No";
486
487         // if TotalPages is greater than currentPage means it has nextPage
488         var nextPage = currentPage < TotalPages ? "Yes" : "No";
489
490         // Object which we are going to send in header
491         var paginationMetadata = new
492     {
```



```
ACController.cs*  MembersController.cs  Source Control Explorer  ACO_MembersMart.cs  Notification_UserSpecific.cs  UQP  UQP.Controllers.MRAController  GetMemberDetails(string reporttype, HealthplanMRA member, PagingPa...  
520  
521  
522  
523     }  
524     else if (reporttype == "totalmembership")  
525     {  
526         dynamic querym = null;  
527         List<LVisitReport> lstmembercount = new List<LVisitReport>();  
528         //string sdate = "31/01/" + DateTime.Now.Year.ToString();  
529  
530         //string edate = "28/" + member.month + "/" + member.year;  
531         //DateTime enddate = DateTime.ParseExact(edate, "dd/MM/yyyy", null);  
532         //DateTime startdate = DateTime.ParseExact(sdate, "dd/MM/yyyy", null);  
533         var today = DateTime.Today;  
534         var month = new DateTime(today.Year, today.Month, 1);  
535         var first = month.AddMonths(-1);  
536         var enddate = month.AddDays(-1);  
537         if (member.report == false)  
538         {  
539             if (reporttype == "totalmembership")  
540             {  
541                 querym = (from m in db.NewReportByMembers.AsNoTracking()  
542                     where (member.HealthplanId == 0 || m.Healthplan_id == member.HealthplanId) && (member.providerid == 0 ||  
543                         m.provider_id == member.providerid)  
544                         && m.PlanEffectiveDate <= enddate && m.PlanTerminationDate >= enddate && m.TimeStamp.Year == enddate.Year  
545                         && m.TimeStamp.Month == enddate.Month  
546                         && (member.dob == null || m.DateofBirth == member.dob)  
547                         && (member.membername == null || m.FirstName.StartsWith(member.membername) || m.LastName.StartsWith(member.membername) ||  
548                         (m.FirstName + " " + m.LastName).StartsWith(member.membername))  
549                         && (member.subsId == null || m.SubscriberID.StartsWith(member.subsId))  
550                         && (string.IsNullOrEmpty(member.status) || m.Status == member.status)  
551  
552  
553  
554  
555
```

A screenshot of a Microsoft Visual Studio IDE interface. The main window shows a C# code editor with the following code:

```
Debug Any CPU IIS Express (Google Chrome) MembersController.cs ACO_MembersMart.cs Notification_UserSpecific.cs GetMemberDetails(string reporttype, HealthplanMRA member, PagingParameterModel pagingParameterModel)
{
    var query = db.NewReportByMembers
        .Where(m => m.SubscriberID.StartsWith(member.subsId))
        .Where(m => string.IsNullOrEmpty(member.status) || m.Status == member.status)
        .Where(m => member.age == 0 || m.DateOfBirth >= DateTime.Today.AddYears(-18))
        .Select(new
        {
            m.SubscriberID,
            m.FirstName,
            m.LastName,
            m.MiddleName,
            m.DateOfBirth,
            m.Gender,
            m.ProviderFirstName,
            m.ProviderMiddleName,
            m.ProviderLastName,
            m.HealthplanName,
            m.Status
        })
        .Distinct()
        .OrderBy(e => new { e.LastName, e.FirstName })
        .Take(pagingParameterModel.pageSize * (pagingParameterModel.pageNumber - 1))
        .Skip(pagingParameterModel.pageSize * (pagingParameterModel.pageNumber - 1));
    totalcount = (from m in db.NewReportByMembers.AsNoTracking()
        where (member.HealthplanId == 0 || m.Healthplan_id == member.HealthplanId) && (member.providerid == 0 || m.provider_id == member.providerid) && m.PlanEffectiveDate <= enddate && m.PlanTerminationDate >= enddate &&
        m.TimeStamp.Year == enddate.Year
        && m.TimeStamp.Month == enddate.Month
        && (member.dob == null || m.DateOfBirth == member.dob)
        && (member.membername == null || m.FirstName.StartsWith(member.membername) || m.LastName.StartsWith(member.membername))
        && (member.membername == null || m.FirstName + " " + m.LastName.StartsWith(member.membername))
        && (member.subsId == null || m.SubscriberID.StartsWith(member.subsId))
        && (string.IsNullOrEmpty(member.status) || m.Status == member.status)
    ).Count();
}
```

The Solution Explorer on the right side shows the project structure for 'UQP'.

JQP - Microsoft Visual Studio

File View Project Build Debug Team Tools Test Analyze Window Help

Debug Any CPU IIS Express (Google Chrome)

RAController.cs MembersController.cs Source Control Explorer ACO\_MembersMart.cs Notification\_UserSpecific.cs

UOP UOP.Controllers.MRAController GetMemberDetails(string reporttype, HealthplanMRA member, PagingPa

```
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
}
else if (member.report == true)
{
    querym = (from m in db.NewReportByMembers.AsNoTracking()
               where (member.HealthplanId == 0 || m.Healthplan_id == member.HealthplanId) && (member.providerid == 0 || m.provider_id == member.providerid) && m.PlanEffectiveDate <= enddate && m.PlanTerminationDate >= enddate && m.TimeStamp.Year == enddate.Year
               && m.TimeStamp.Month == enddate.Month
               && (member.dob == null || m.DateOfBirth == member.dob)
               && (member.membername == null || m.FirstName.StartsWith(member.membername) || m.LastName.StartsWith(member.membername) ||
                     (m.FirstName + " " + m.LastName).StartsWith(member.membername))
               && (member.subsId == null || m.SubscriberID.StartsWith(member.subsId))
               && (member.status == null || m.Status == member.status))
    select new
    {
        m.SubscriberID,
        m.FirstName,
        m.LastName,
        m.MiddleName,
        m.DateOfBirth,
        m.Gender,
        m.ProviderFirstName,
        m.ProviderMiddleName,
        m.ProviderLastName,
        m.HealthplanName,
        m.Status
    }
).Distinct().OrderBy(e => new { e.LastName, e.FirstName }).ToList();
}
```

Scanned with CamScanner

```
 619
 620
 621
 622
 623
 624
 625
 626
 627
 628
 629
 630
 631
 632
 633
 634
 635
 636
 637
 638
 639
 640
 641
 642
 643
 644
 645
 646
 647
 648
 649
 650
 651
 652
 653
 654
 655
 656
 657
 658
 659
 660
 661
 662
 663
 664
 665
 666
 667
 668
 669
 670
 671
 672
 673
 674
 675
 676
 677
 678
 679
 680
 681
 682
 683
 684
 685
 686
 687
 688
 689
 690
 691
 692
 693
 694
 695
 696
 697
 698
 699
 700
 701
 702
 703
 704
 705
 706
 707
 708
 709
 710
 711
 712
 713
 714
 715
 716
 717
 718
 719
 720
 721
 722
 723
 724
 725
 726
 727
 728
 729
 730
 731
 732
 733
 734
 735
 736
 737
 738
 739
 740
 741
 742
 743
 744
 745
 746
 747
 748
 749
 750
 751
 752
 753
 754
 755
 756
 757
 758
 759
 760
 761
 762
 763
 764
 765
 766
 767
 768
 769
 770
 771
 772
 773
 774
 775
 776
 777
 778
 779
 780
 781
 782
 783
 784
 785
 786
 787
 788
 789
 790
 791
 792
 793
 794
 795
 796
 797
 798
 799
 800
 801
 802
 803
 804
 805
 806
 807
 808
 809
 810
 811
 812
 813
 814
 815
 816
 817
 818
 819
 820
 821
 822
 823
 824
 825
 826
 827
 828
 829
 830
 831
 832
 833
 834
 835
 836
 837
 838
 839
 840
 841
 842
 843
 844
 845
 846
 847
 848
 849
 850
 851
 852
 853
 854
 855
 856
 857
 858
 859
 860
 861
 862
 863
 864
 865
 866
 867
 868
 869
 870
 871
 872
 873
 874
 875
 876
 877
 878
 879
 880
 881
 882
 883
 884
 885
 886
 887
 888
 889
 890
 891
 892
 893
 894
 895
 896
 897
 898
 899
 900
 901
 902
 903
 904
 905
 906
 907
 908
 909
 910
 911
 912
 913
 914
 915
 916
 917
 918
 919
 920
 921
 922
 923
 924
 925
 926
 927
 928
 929
 930
 931
 932
 933
 934
 935
 936
 937
 938
 939
 940
 941
 942
 943
 944
 945
 946
 947
 948
 949
 950
 951
 952
 953
 954
 955
 956
 957
 958
 959
 960
 961
 962
 963
 964
 965
 966
 967
 968
 969
 970
 971
 972
 973
 974
 975
 976
 977
 978
 979
 980
 981
 982
 983
 984
 985
 986
 987
 988
 989
 990
 991
 992
 993
 994
 995
 996
 997
 998
 999
 1000
 1001
 1002
 1003
 1004
 1005
 1006
 1007
 1008
 1009
 1010
 1011
 1012
 1013
 1014
 1015
 1016
 1017
 1018
 1019
 1020
 1021
 1022
 1023
 1024
 1025
 1026
 1027
 1028
 1029
 1030
 1031
 1032
 1033
 1034
 1035
 1036
 1037
 1038
 1039
 1040
 1041
 1042
 1043
 1044
 1045
 1046
 1047
 1048
 1049
 1050
 1051
 1052
 1053
 1054
 1055
 1056
 1057
 1058
 1059
 1060
 1061
 1062
 1063
 1064
 1065
 1066
 1067
 1068
 1069
 1070
 1071
 1072
 1073
 1074
 1075
 1076
 1077
 1078
 1079
 1080
 1081
 1082
 1083
 1084
 1085
 1086
 1087
 1088
 1089
 1090
 1091
 1092
 1093
 1094
 1095
 1096
 1097
 1098
 1099
 1100
 1101
 1102
 1103
 1104
 1105
 1106
 1107
 1108
 1109
 1110
 1111
 1112
 1113
 1114
 1115
 1116
 1117
 1118
 1119
 1120
 1121
 1122
 1123
 1124
 1125
 1126
 1127
 1128
 1129
 1130
 1131
 1132
 1133
 1134
 1135
 1136
 1137
 1138
 1139
 1140
 1141
 1142
 1143
 1144
 1145
 1146
 1147
 1148
 1149
 1150
 1151
 1152
 1153
 1154
 1155
 1156
 1157
 1158
 1159
 1160
 1161
 1162
 1163
 1164
 1165
 1166
 1167
 1168
 1169
 1170
 1171
 1172
 1173
 1174
 1175
 1176
 1177
 1178
 1179
 1180
 1181
 1182
 1183
 1184
 1185
 1186
 1187
 1188
 1189
 1190
 1191
 1192
 1193
 1194
 1195
 1196
 1197
 1198
 1199
 1200
 1201
 1202
 1203
 1204
 1205
 1206
 1207
 1208
 1209
 1210
 1211
 1212
 1213
 1214
 1215
 1216
 1217
 1218
 1219
 1220
 1221
 1222
 1223
 1224
 1225
 1226
 1227
 1228
 1229
 1230
 1231
 1232
 1233
 1234
 1235
 1236
 1237
 1238
 1239
 1240
 1241
 1242
 1243
 1244
 1245
 1246
 1247
 1248
 1249
 1250
 1251
 1252
 1253
 1254
 1255
 1256
 1257
 1258
 1259
 1260
 1261
 1262
 1263
 1264
 1265
 1266
 1267
 1268
 1269
 1270
 1271
 1272
 1273
 1274
 1275
 1276
 1277
 1278
 1279
 1280
 1281
 1282
 1283
 1284
 1285
 1286
 1287
 1288
 1289
 1290
 1291
 1292
 1293
 1294
 1295
 1296
 1297
 1298
 1299
 1300
 1301
 1302
 1303
 1304
 1305
 1306
 1307
 1308
 1309
 1310
 1311
 1312
 1313
 1314
 1315
 1316
 1317
 1318
 1319
 1320
 1321
 1322
 1323
 1324
 1325
 1326
 1327
 1328
 1329
 1330
 1331
 1332
 1333
 1334
 1335
 1336
 1337
 1338
 1339
 1340
 1341
 1342
 1343
 1344
 1345
 1346
 1347
 1348
 1349
 1350
 1351
 1352
 1353
 1354
 1355
 1356
 1357
 1358
 1359
 1360
 1361
 1362
 1363
 1364
 1365
 1366
 1367
 1368
 1369
 1370
 1371
 1372
 1373
 1374
 1375
 1376
 1377
 1378
 1379
 1380
 1381
 1382
 1383
 1384
 1385
 1386
 1387
 1388
 1389
 1390
 1391
 1392
 1393
 1394
 1395
 1396
 1397
 1398
 1399
 1400
 1401
 1402
 1403
 1404
 1405
 1406
 1407
 1408
 1409
 1410
 1411
 1412
 1413
 1414
 1415
 1416
 1417
 1418
 1419
 1420
 1421
 1422
 1423
 1424
 1425
 1426
 1427
 1428
 1429
 1430
 1431
 1432
 1433
 1434
 1435
 1436
 1437
 1438
 1439
 1440
 1441
 1442
 1443
 1444
 1445
 1446
 1447
 1448
 1449
 1450
 1451
 1452
 1453
 1454
 1455
 1456
 1457
 1458
 1459
 1460
 1461
 1462
 1463
 1464
 1465
 1466
 1467
 1468
 1469
 1470
 1471
 1472
 1473
 1474
 1475
 1476
 1477
 1478
 1479
 1480
 1481
 1482
 1483
 1484
 1485
 1486
 1487
 1488
 1489
 1490
 1491
 1492
 1493
 1494
 1495
 1496
 1497
 1498
 1499
 1500
 1501
 1502
 1503
 1504
 1505
 1506
 1507
 1508
 1509
 1510
 1511
 1512
 1513
 1514
 1515
 1516
 1517
 1518
 1519
 1520
 1521
 1522
 1523
 1524
 1525
 1526
 1527
 1528
 1529
 1530
 1531
 1532
 1533
 1534
 1535
 1536
 1537
 1538
 1539
 1540
 1541
 1542
 1543
 1544
 1545
 1546
 1547
 1548
 1549
 1550
 1551
 1552
 1553
 1554
 1555
 1556
 1557
 1558
 1559
 1560
 1561
 1562
 1563
 1564
 1565
 1566
 1567
 1568
 1569
 1570
 1571
 1572
 1573
 1574
 1575
 1576
 1577
 1578
 1579
 1580
 1581
 1582
 1583
 1584
 1585
 1586
 1587
 1588
 1589
 1590
 1591
 1592
 1593
 1594
 1595
 1596
 1597
 1598
 1599
 1600
 1601
 1602
 1603
 1604
 1605
 1606
 1607
 1608
 1609
 1610
 1611
 1612
 1613
 1614
 1615
 1616
 1617
 1618
 1619
 1620
 1621
 1622
 1623
 1624
 1625
 1626
 1627
 1628
 1629
 1630
 1631
 1632
 1633
 1634
 1635
 1636
 1637
 1638
 1639
 1640
 1641
 1642
 1643
 1644
 1645
 1646
 1647
 1648
 1649
 1650
 1651
 1652
 1653
 1654
 1655
 1656
 1657
 1658
 1659
 1660
 1661
 1662
 1663
 1664
 1665
 1666
 1667
 1668
 1669
 1670
 1671
 1672
 1673
 1674
 1675
 1676
 1677
 1678
 1679
 1680
 1681
 1682
 1683
 1684
 1685
 1686
 1687
 1688
 1689
 1690
 1691
 1692
 1693
 1694
 1695
 1696
 1697
 1698
 1699
 1700
 1701
 1702
 1703
 1704
 1705
 1706
 1707
 1708
 1709
 1710
 1711
 1712
 1713
 1714
 1715
 1716
 1717
 1718
 1719
 1720
 1721
 1722
 1723
 1724
 1725
 1726
 1727
 1728
 1729
 1729
 1730
 1731
 1732
 1733
 1734
 1735
 1736
 1737
 1738
 1739
 1739
 1740
 1741
 1742
 1743
 1744
 1745
 1746
 1747
 1748
 1749
 1749
 1750
 1751
 1752
 1753
 1754
 1755
 1756
 1757
 1758
 1759
 1759
 1760
 1761
 1762
 1763
 1764
 1765
 1766
 1767
 1768
 1769
 1769
 1770
 1771
 1772
 1773
 1774
 1775
 1776
 1777
 1778
 1779
 1779
 1780
 1781
 1782
 1783
 1784
 1785
 1786
 1787
 1788
 1789
 1789
 1790
 1791
 1792
 1793
 1794
 1795
 1796
 1797
 1798
 1799
 1799
 1800
 1801
 1802
 1803
 1804
 1805
 1806
 1807
 1808
 1809
 1809
 1810
 1811
 1812
 1813
 1814
 1815
 1816
 1817
 1818
 1819
 1819
 1820
 1821
 1822
 1823
 1824
 1825
 1826
 1827
 1828
 1829
 1829
 1830
 1831
 1832
 1833
 1834
 1835
 1836
 1837
 1838
 1839
 1839
 1840
 1841
 1842
 1843
 1844
 1845
 1846
 1847
 1848
 1849
 1849
 1850
 1851
 1852
 1853
 1854
 1855
 1856
 1857
 1858
 1859
 1859
 1860
 1861
 1862
 1863
 1864
 1865
 1866
 1867
 1868
 1869
 1869
 1870
 1871
 1872
 1873
 1874
 1875
 1876
 1877
 1878
 1879
 1879
 1880
 1881
 1882
 1883
 1884
 1885
 1886
 1887
 1888
 1889
 1889
 1890
 1891
 1892
 1893
 1894
 1895
 1896
 1897
 1898
 1899
 1899
 1900
 1901
 1902
 1903
 1904
 1905
 1906
 1907
 1908
 1909
 1909
 1910
 1911
 1912
 1913
 1914
 1915
 1916
 1917
 1918
 1919
 1919
 1920
 1921
 1922
 1923
 1924
 1925
 1926
 1927
 1928
 1929
 1929
 1930
 1931
 1932
 1933
 1934
 1935
 1936
 1937
 1938
 1939
 1939
 1940
 1941
 1942
 1943
 1944
 1945
 1946
 1947
 1948
 1949
 1949
 1950
 1951
 1952
 1953
 1954
 1955
 1956
 1957
 1958
 1959
 1959
 1960
 1961
 1962
 1963
 1964
 1965
 1966
 1967
 1968
 1969
 1969
 1970
 1971
 1972
 1973
 1974
 1975
 1976
 1977
 1978
 1979
 1979
 1980
 1981
 1982
 1983
 1984
 1985
 1986
 1987
 1988
 1989
 1989
 1990
 1991
 1992
 1993
 1994
 1995
 1996
 1997
 1998
 1999
 1999
 2000
 2001
 2002
 2003
 2004
 2005
 2006
 2007
 2008
 2009
 2009
 2010
 2011
 2012
 2013
 2014
 2015
 2016
 2017
 2018
 2019
 2019
 2020
 2021
 2022
 2023
 2024
 2025
 2026
 2027
 2028
 2029
 2029
 2030
 2031
 2032
 2033
 2034
 2035
 2036
 2037
 2038
 2039
 2039
 2040
 2041
 2042
 2043
 2044
 2045
 2046
 2047
 2048
 2049
 2049
 2050
 2051
 2052
 2053
 2054
 2055
 2056
 2057
 2058
 2059
 2059
 2060
 2061
 2062
 2063
 2064
 2065
 2066
 2067
 2068
 2069
 2069
 2070
 2071
 2072
 2073
 2074
 2075
 2076
 2077
 2078
 2079
 2079
 2080
 2081
 2082
 2083
 2084
 2085
 2086
 2087
 2088
 2089
 2089
 2090
 2091
 2092
 2093
 2094
 2095
 2096
 2097
 2098
 2099
 2099
 2100
 2101
 2102
 2103
 2104
 2105
 2106
 2107
 2108
 2109
 2109
 2110
 2111
 2112
 2113
 2114
 2115
 2116
 2117
 2118
 2119
 2119
 2120
 2121
 2122
 2123
 2124
 2125
 2126
 2127
 2128
 2129
 2129
 2130
 2131
 2132
 2133
 2134
 2135
 2136
 2137
 2138
 2139
 2139
 2140
 2141
 2142
 2143
 2144
 2145
 2146
 2147
 2148
 2149
 2149
 2150
 2151
 2152
 2153
 2154
 2155
 2156
 2157
 2158
 2159
 2159
 2160
 2161
 2162
 2163
 2164
 2165
 2166
 2167
 2168
 2169
 2169
 2170
 2171
 2172
 2173
 2174
 2175
 2176
 2177
 2178
 2179
 2179
 2180
 2181
 2182
 2183
 2184
 2185
 2186
 2187
 2188
 2189
 2189
 2190
 2191
 2192
 2193
 2194
 2195
 2196
 2197
 2198
 2199
 2199
 2200
 2201
 2202
 2203
 2204
 2205
 2206
 2207
 2208
 2209
 2209
 2210
 2211
 2212
 2213
 2214
 2215
 2216
 2217
 2218
 2219
 2219
 2220
 2221
 2222
 2223
 2224
 2225
 2226
 2227
 2228
 2229
 2229
 2230
 2231
 2232
 2233
 2234
 2235
 2236
 2237
 2238
 2239
 2239
 2240
 2241
 2242
 2243
 2244
 2245
 2246
 2247
 2248
 2249
 2249
 2250
 2251
 2252
 2253
 2254
 2255
 2256
 2257
 2258
 2259
 2259
 2260
 2261
 2262
 2263
 2264
 2265
 2266
 2267
 2268
 2269
 2269
 2270
 2271
 2272
 2273
 2274
 2275
 2276
 2277
 2278
 2279
 2279
 2280
 2281
 2282
 2283
 2284
 2285
 2286
 2287
 2288
 2289
 2289
 2290
 2291
 2292
 2293
 2294
 2295
 2296
 2297
 2298
 2299
 2299
 2300
 2301
 2302
 2303
 2304
 2305
 2306
 2307
 2308
 2309
 2309
 2310
 2311
 2312
 2313
 2314
 2315
 2316
 2317
 2318
 2319
 2319
 2320
 2321
 2322
 2323
 2324
 2325
 2326
 2327
 2328
 2329
 2329
 2330
 2331
 2332
 2333
 2334
 2335
 2336
 2337
 2338
 2339
 2339
 2340
 2341
 2342
 2343
 2344
 2345
 2346
 2347
 2348
 2349
 2349
 2350
 2351
 2352
 2353
 2354
 2355
 2356
 2357
 2358
 2359
 2359
 2360
 2361
 2362
 2363
 2364
 2365
 2366
 2367
 2368
 2369
 2369
 2370
 2371
 2372
 2373
 2374
 2375
 2376
 2377
 2378
 2379
 2379
 2380
 2381
 2382
 2383
 2384
 2385
 2386
 2387
 2388
 2389
 2389
 2390
 2391
 2392
 2393
 2394
 2395
 2396
 2397
 2398
 2399
 2399
 2400
 2401
 2402
 2403
 2404
 2405
 2406
 2407
 2408
 2409
 2409
 2410
 2411
 2412
 2413
 2414
 2415
 2416
 2417
 2418
 2419
 2419
 2420
 2421
 2422
 2423
 2424
 2425
 2426
 2427
 2428
 2429
 2429
 2430
 2431
 2432
 2433
 2434
 2435
 2436
 2437
 2438
 2439
 2439
 2440
 2441
 2442
 2443
 2444
 2445
 2446
 2447
 2448
 2449
 2449
 2450
 2451
 2452
 2453
 2454
 2455
 2456
 2457
 2458
 2459
 2459
 2460
 2461
 2462
 2463
 2464
 2465
```

```
ACController.cs*  X MembersController.cs  Source Control Explorer  ACO_MemberDetails  Notification_ScheduleDetails  UQP  GetMemberDetails(string reporttype, HealthplanMRA member)
646 } }
647 #region pagination
648 if (member.report == false)
649 {
650     // Get's No of Rows Count
651     int count = totalcount;
652     //lstmembercount.Count();
653     // int count = 0;
654
655     // Parameter is passed from Query string if it is null then it default Value will be pageNumber:1
656     int CurrentPage = pagingParameterModel.pageNumber;
657
658     // Parameter is passed from Query string if it is null then it default Value will be pageSize:20
659     int PageSize = pagingParameterModel.pageSize;
660
661     // Display TotalCount to Records to User
662     int TotalCount = count;
663
664     // Calculating Totalpage by Dividing (No of Records / Pagesize)
665     int TotalPages = (int)Math.Ceiling(count / (double)PageSize);
666
667     // Returns List of Customer after applying Paging
668     var items = lstmembercount;//.Skip((CurrentPage - 1) * PageSize).Take(PageSize).ToList();
669
670     // if CurrentPage is greater than 1 means it has previousPage
671     var previousPage = currentPage > 1 ? "Yes" : "No";
672
673     // if TotalPages is greater than currentPage means it has nextPage
674     var nextPage = currentPage < TotalPages ? "Yes" : "No";
675
676     // Object which we are going to send in header
677     var paginationMetadata = new
678     {
679
680
681 }
```

```
UOP.Controllers.MRAController.cs  MembersController.cs  Source Control Explorer  ACO_MembersMart.cs  Notification_UserSpecific.cs  GetMemberDetails(string reportype, HealthplanMRA member, string zipFileName)
UOP
679 // Object which we are going to send in header
680 var paginationMetadata = new
681 {
682     totalCount = TotalCount,
683     pageSize = PageSize,
684     currentPage = CurrentPage,
685     totalPages = TotalPages,
686     previousPage,
687     nextPage
688 };
689 // Setting Header
690 System.Web.HttpContext.Current.Response.Headers.Add("Paging-Headers", Newtonsoft.Json.JsonConvert.SerializeObject(paginationMetadata));
691 //HttpContext.Current.Response.Headers.Add("Filename", zipFileName);
692 HttpContext.Current.Response.Headers.Add("Access-Control-Expose-Headers", "Paging-Headers");
693 return Request.CreateResponse(HttpStatusCode.OK, items);
694 }
695 #endregion
696 else if (member.report == true)
697 {
698     var rpoj = new ReportsController
699     {
700         Request = new System.Net.Http.HttpRequestMessage(),
701         Configuration = new HttpConfiguration()
702     };
703     return rpoj.GenerateReport(null, null, lstmembercount, "MRA Total Membership", "excel", null, null, null, null);
704 }
705 }
706 }
707 }
708 }
709 }
710 }
711 }
712 }
713 }
714 }
catch (Exception ex)
{
    
```

```
UQP UQP.Controllers.MRAController GetMemberDetails(string reporttype, HealthplanMRA member)
715     return Request.CreateResponse(HttpStatusCode.BadRequest, ex);
716 }
717 }
718 }
719 }
720 [HttpPost]
721 [Route("api/mra/MissingConditions/{conditiontype}")]
722     public HttpResponseMessage GetMissingConditions(HealthplanMRA hpMRA, string conditiontype)
723     {
724
725         try
726     {
727         using (UQPEntities db = new UQPEntities(HttpContext.Current.Request.Headers.GetValues("IPA").FirstOrDefault()))
728     {
729         var mrahp =
730             (from mra in db.MRA_InsuranceMissingConditions.AsNoTracking()
731             where
732                 (conditiontype == "CurrentMissingConditions" ? mra.year == (hpMRA.year - 1) : mra.year == hpMRA.year) &&
733                 mra.HealthplanId == hpMRA.HealthplanId
734             select new { mra.MissingConditionID, mra.HealthplanId, mra.HealthplanName, mra.MissingConditions }).Distinct();
735         var x = mrahp.Count();
736
737         List<MRA_InsuranceMissingConditions> mramissinglist = new List<MRA_InsuranceMissingConditions>();
738
739         foreach (var per in mrahp)
740     {
741             MRA_InsuranceMissingConditions mrahpobj = new MRA_InsuranceMissingConditions();
742             mrahpobj.HealthplanId = per.HealthplanId;
743             mrahpobj.HealthplanName = per.HealthplanName;
744             mrahpobj.MissingConditions = per.MissingConditions;
745             mrahpobj.MissingConditionID = per.MissingConditionID;
746
747             mramissinglist.Add(mrahpobj);
748
749     }
    }
}
100 %
```

Output

Show output from Source Control - Team Foundation

UQP

UQP.Controllers.MRAController

GetMemberDetails(string reporttype, Healthplan

```
732     (conditiontype == "CurrentMissingConditions" ? mra.year == (hpMRA.year - 1) : mra.year == hpMRA.year) &&
733     mra.HealthplanId == hpMRA.HealthplanId
734     select new { mra.MissingConditionID, mra.HealthplanId, mra.HealthplanName, mra.MissingConditions }).Distinct();
735     var x = mrahp.Count();
736
737     List<MRA_InsuranceMissingConditions> mramissinglist = new List<MRA_InsuranceMissingConditions>();
738
739     foreach (var per in mrahp)
740     {
741         MRA_InsuranceMissingConditions mrahpobj = new MRA_InsuranceMissingConditions();
742         mrahpobj.HealthplanId = per.HealthplanId;
743         mrahpobj.HealthplanName = per.HealthplanName;
744         mrahpobj.MissingConditions = per.MissingConditions;
745         mrahpobj.MissingConditionID = per.MissingConditionID;
746
747         mramissinglist.Add(mrahpobj);
748
749
750     };
751     return Request.CreateResponse(mramissinglist);
752 }
753
754
755 }
756 catch (Exception e)
757 {
758     return Request.CreateResponse(e.InnerException);
759 }
760 }
```

```
 765
 766
 767
 768     [HttpPost]
 769     [Route("api/mra/GetMRAGaps/")]
 770     public HttpResponseMessage GetMRAGaps([FromUri]HealthplanMRA hpMRA, [FromUri]PagingParameterModel pagingparametermodel, int[] ids)
 771     {
 772
 773         try
 774         {
 775             using (UQPEntities db = new UQPEntities(HttpContext.Current.Request.Headers.GetValues("IPA").FirstOrDefault()))
 776             {
 777                 db.Database.CommandTimeout = 240;
 778                 // dynamic mrahp = null;
 779                 int totalcount = 0;
 780
 781
 782
 783                 totalcount = (from mra in db.MRA_MembersMissingConditions
 784                             where mra.IsPresentInMM == true &&
 785 (hpMRA.conditiontype == "CurrentMissingConditions" ? mra.PROjectedScoreYear == (hpMRA.year - 1) : mra.PROjectedScoreYear == hpMRA.year) &&
 786 : mra.InServicefund == null || mra.InGateway == null) &&
 787 // (mra.HealthplanId == 0 || mra.HealthPlanID == hpMRA.HealthplanId)
 788 // (hpMRA.providerid == 0 || mra.provider_id == hpMRA.providerid) &&
 789 && (hpMRA.gapstatus == null || mra.Status == hpMRA.gapstatus.Value)
 790 (hpMRA.subsId == null || mra.RosterSubscriberId.StartsWith(hpMRA.subsId))
 791 && (hpMRA.membername == null || mra.SubscriberFirstName.StartsWith(hpMRA.membername) || mra.SubscriberLastName.StartsWith(hpMRA.membername)) ||
 792 && (hpMRA.membername == null || mra.SubscriberFirstName + " " + mra.SubscriberLastName).StartsWith(hpMRA.membername))
 793 (mra.SubscriberFirstName + " " + mra.SubscriberLastName).StartsWith(hpMRA.membername)) && (mra.MissingConditionID != null)
 794 && (ids.Contains((int)mra.MissingConditionID)) && (mra.MissingConditionID != null)
 795
 796
 797                 select new { mra.RosterSubscriberId, mra.MissingConditionID }).Distinct().Count();
 798
 799             dynamic mrahp = null;
```

The screenshot shows a Microsoft Visual Studio interface with the following details:

- Project Explorer:** Shows the solution structure with files like MRAController.cs, MembersController.cs, Source Control Explorer, ACO\_MembersMart.cs, Notification\_UserSpecific.cs, and GetMRA Gaps.
- Code Editor:** The main window displays the `UQP.Controller.MRAController.cs` file. The code is a C# class containing methods for querying MRA data. It includes logic for filtering by subscriber names, handling report generation, and selecting specific fields from the database.
- Status Bar:** Shows "100%" and "Output".
- Output Window:** Displays the message "Show output from: Source Control - Team Foundation".

```
794     (mra.SubscriberFirstName + " " + mra.SubscriberLastName).StartsWith(hpMRA.membername))
795     && (ids.Contains((int)mra.MissingConditionID)) && (mra.MissingConditionID != null)
796         select new { mra.RosterSubscriberid, mra.MissingConditionID }).Distinct().Count();
797
798     dynamic mrahp = null;
799
800
801
802     if (hpMRA.report == true)
803     {
804         pagingparametermodel.pageNumber = 1;
805         pagingparametermodel.pageSize = 2 * totalcount;
806     }
807
808
809     mrahp = (from mra in db.MRA_MembersMissingConditions
810             where mra.IsPresentInMM == true &&
811 (hpMRA.conditiontype == "CurrentMissingConditions" ? mra.ProjectedScoreYear == (hpMRA.year - 1)
812 : mra.PProjectedScoreYear == hpMRA.year)
813 //&& (mra.InServicefund == null || mra.InGateway == null)
814 &&
815 (hpMRA.HealthplanId == 0 || mra.HealthPlanId == hpMRA.HealthplanId) && (hpMRA.providerid == 0 || mra.provider_id == hpMRA.providerid) &&
816 (hpMRA.gapstatus == null || mra.Status == hpMRA.gapstatus.Value)
817 && (hpMRA.subsId == null || mra.RosterSubscriberid.StartsWith(hpMRA.subsId))
818     && (hpMRA.membername == null || mra.SubscriberFirstName.StartsWith(hpMRA.membername) || mra.SubscriberLastName.StartsWith(hpMRA.membername))
819     ||
820     (mra.SubscriberFirstName + " " + mra.SubscriberLastName).StartsWith(hpMRA.membername)
821     && ids.Contains((int)mra.MissingConditionID)
822
823
824
825
826
827
828
829     select new
830     {
831         mra.HealthPlanID,
832         mra.HealthPlanName,
833         mra.RosterSubscriberid,
834         mra.SubscriberFirstName,
835         mra.SubscriberIdName,
836         mra.SubscriberLastName
837     }
```

UQP - Microsoft Visual Studio

File Edit View Project Build Debug Team Tools Test Analyze Window Help

Debug Any CPU IIS Express (Google Chrome)

MRAController.cs MembersController.cs Source Control Explorer ACO\_MembersMart.cs Notification\_UserSpecific.cs GetMRA Gaps(HealthplanMRA hpMRA, PagingParameterModel pagingpa

UQP

```
827 mra.SubscriberFirstName,
828 mra.SubscriberMidName,
829 mra.SubscriberLastName
830
831 mra.age,
832 mra.MissingConditionID,
833 mra.HCCcategoryDescription,
834 provfirstname = mra.providerfirstname,
835 provmidname = mra.providermidname,
836 provlastname = mra.providerlastname,
837 mra.Status,
838 mra.DOB
839
840 }).Distinct().OrderBy(e => new { e.SubscriberLastName, e.SubscriberFirstName })
841 .Skip(pagingparametermodel.pageSize * (pagingparametermodel.pageNumber - 1)).Take(pagingparametermodel.pageSize).ToList();
842
843 List<MRA_MembersMissingConditions> mrmissinglist = new List<MRA_MembersMissingConditions>();
844
845 foreach (var per in mrhobj)
846 {
847     MRA_MembersMissingConditions mrhobj = new MRA_MembersMissingConditions();
848     mrhobj.HealthPlanID = per.HealthPlanID;
849     mrhobj.HealthPlanName = per.HealthPlanName;
850     mrhobj.HCCcategoryDescription = per.HCCcategoryDescription;
851     mrhobj.MissingConditionID = per.MissingConditionID;
852     mrhobj.SubscriberFirstName = per.SubscriberFirstName;
853     //mrhobj.SubscriberMidName = per.SubscriberMidName;
854     mrhobj.SubscriberLastName = per.SubscriberLastName;
855     mrhobj.RosterSubscriberid = per.RosterSubscriberid;
856     mrhobj.age = per.age;
857     mrhobj.MissingConditionID = per.MissingConditionID;
858     mrhobj.providerfirstname = per.provfirstname;
859     //mrhobj.providermidname = per.provmidname;
860     mrhobj.providerlastname = (per.provlastname == null || per.provlastname == "") ? "NA" : per.provlastname;
861     mrhobj.Status = per.Status;
862 }
```

Output

Show output from: Source Control - Team Foundation

Solution Explorer

View Project Build Debug Team Tools Test Analyze Window Help

Debug Any CPU IIS Express (Google Chrome)

MRAController.cs\* MembersController.cs Source Control Explorer ACO\_MembersMart.cs Notification\_UserSpecific.cs

UQP

```
857     mrahpobj.age = per.age;
858     mrahpobj.MissingConditionID = per.MissingConditionID;
859     mrahpobj.providerfirstname = per.provfirstname;
860     //mrahpobj.providermidname = per.provmidname;
861     mrahpobj.providerlastname = (per.provlastname == null || per.provlastname == "") ? "NA" : per.provlastname;
862     mrahpobj.Status = per.Status;
863     mrahpobj.DOB = per.DOB;

864
865     mramissinglist.Add(mrahpobj);
866
867 }
868
869 if (hpMRA.report == false)
870 {
871     int count = totalcount;
872
873     int CurrentPage = pagingparametermodel.pageNumber;
874     int PageSize = pagingparametermodel.pageSize;
875     int TotalCount = count;
876     int TotalPages = (int)Math.Ceiling(count / (double)PageSize);

877
878     var items = mramissinglist;
879
880     var previousPage = CurrentPage > 1 ? "Yes" : "No";
881
882     var nextPage = CurrentPage < TotalPages ? "Yes" : "No";
883
884
885     var paginationMetadata = new
886     {
887         totalCount = TotalCount,
888         pageSize = PageSize,
889     };
890
891
892 }
```

The screenshot shows a Microsoft Visual Studio interface with the following details:

- Open Files:** MRAController.cs, MembersController.cs, Source Control Explorer, ACO\_MembersMart.cs, Notification\_UserSpecific.cs, and UQP.Controllers.MRAController.
- Current File:** UQP.Controllers.MRAController
- Code Snippet:** The code is a C# class definition for MRAController. It includes logic for generating reports based on parameters. The snippet shows methods for handling requests and generating responses, including the use of Newtonsoft.Json and System.Web.HttpContext.
- Output Window:** Shows "Output" and "Show output from: Source Control - Team Foundation".
- Solution Explorer:** Shows a tree view of the solution structure.

```
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925 };
```

```
totalCount = TotalCount,
pageSize = PageSize,
currentPage = currentPage,
totalPages = TotalPages,
previousPage,
nextPage
```

```
};

System.Web.HttpContext.Current.Response.Headers.Add("Paging-Headers", Newtonsoft.Json.JsonConvert.SerializeObject(paginationMetadata));

HttpContext.Current.Response.Headers.Add("Access-Control-Expose-Headers", "Paging-Headers");
return Request.CreateResponse(HttpStatusCode.OK, items);
}
else if (hpMRA.report == true)
{
    var rpoj = new ReportsController
    {
        Request = new System.Net.Http.HttpRequestMessage(),
        Configuration = new HttpConfiguration()
    };
    return rpoj.GenerateNotifReport(null, null, null, null, null, null, (hpMRA.conditiontype == "CurrentMissingCondition") ?
        "Current Gaps" : "Projected Gaps", "excel", mramissinglist, null, null);
}
return Request.CreateResponse(HttpStatusCode.OK, "OK");
}
catch (Exception e)
{
    return Request.CreateResponse(HttpStatusCode.InternalServerError, e.InnerException);
}
```

UQP - Microsoft Visual Studio

Edit View Project Build Debug Team Tools Test Analyze Window Help

Debug Any CPU IIS Express (Google Chrome)

MRAController.cs MembersController.cs ACO\_MembersMart.cs Notification\_UserSpecific.cs

UQP

```
926 [HttpGet]
927 [Route("api/mra/providers")]
928 References
929 public HttpResponseMessage providerpiechartlisting([FromUri]PagingParameterModel pagingParameterModel, [FromUri]HealthplanMRA member)
930 {
931     try
932     {
933         using (UQPEntities db = new UQPEntities(HttpContext.Current.Request.Headers.GetValues("IPA").FirstOrDefault()))
934         {
935             var sortvariable = (member.sort_membership == null && member.sort_score == null) ? "ProviderLastName" :
936                 (member.sort_membership != null) ? "ProjectedMemberCount" + " " + member.sort_membership : "ProjectedScore" + " " + member.sort_score;
937             int totalcount = (member.report == false) ? (from p in db.MRA_ReportByProviderInsurance
938                 where p.Projectedscore >= member.minscore && p.Projectedscore < member.maxscore
939                 && p.ProjectedscoreYear == member.year &&
940                 (member.HealthplanId == 0 || p.HealthplanId == member.HealthplanId)
941                 && (member.membername == null ||
942                     (p.ProviderFirstName.StartsWith(member.membername)) ||
943                     p.ProviderLastName.StartsWith(member.membername)))
944                 select p.ProviderID).Count()
945             : 0;
946
947
948             var query = (member.report == false) ? (from p in db.MRA_ReportByProviderInsurance
949                 where p.Projectedscore >= member.minscore && p.Projectedscore < member.maxscore
950                 && p.ProjectedscoreYear == member.year &&
951                 (member.HealthplanId == 0 || p.HealthplanId == member.HealthplanId)
952                 && (member.membername == null ||
953                     (p.ProviderFirstName.StartsWith(member.membername)) ||
954                     p.ProviderLastName.StartsWith(member.membername)))
955                 select new
956                 {
957                     p.ProviderFirstName,
958                     p.ProviderMidName,
959                     p.ProviderLastName,
960                     member.membername
961                 };
962         }
963     }
964 }
```

Output

Show output from: Source Control - Team Foundation

Solution Explorer

Search Solution Explorer

Solution 'UQP'

- UQP
- Properties
- References
- Action
- API Log
- App Data
- App\_Stat
- bin
- Controllers
  - ACO
  - Audit
  - Bulk
  - CCCDA
  - Dash
  - FileU
  - Guide
  - Meas
  - Mem
  - MRA
  - Notif
  - Proc
  - Prov
  - Repor
  - Resett
  - Star
  - Surve
  - Trend
  - User
  - User\_B
- GapDoc
- LogHelp

UQP - Microsoft Visual Studio

192.61.99.225

Debug Any CPU IIS Express (Google Chrome)

MRAController.cs MembersController.cs Source Control Explorer ACO\_MembersMart.cs Notification\_UserSpecific.cs providerpiechartlisting(PagingParameterModel pagingParameterModel, I

```
UQP
958     p.ProviderFirstName,
959     p.ProviderMidName,
960     p.ProviderLastName,
961     p.HealthplanName,
962     p.Projectedscore,
963     p.ProjectedsMemberCount,
964     p.HealthplanId,
965     p.ProviderID
966   }).Distinct().OrderBy(sortvariable).ToList()
967   .Skip(pagingParameterModel.pageSize * (pagingParameterModel.pageNumber - 1)).Take(pagingParameterModel.pageSize)
968   :
969   (from p in db.MRA_ReportByProviderInsurance
970   where p.Projectedscore >= member.minscore && p.Projectedscore < member.maxscore
971   && p.ProjectedscoreYear == member.year &&
972   (member.HealthplanId == 0 || p.HealthplanId == member.HealthplanId)
973   && (member.membername == null ||
974   (p.ProviderFirstName.StartsWith(member.membername) || p.ProviderLastName.StartsWith(member.membername)))
975   select new
976   {
977     p.ProviderFirstName,
978     p.ProviderMidName,
979     p.ProviderLastName,
980     p.HealthplanName,
981     p.Projectedscore,
982     p.ProjectedsMemberCount,
983     p.HealthplanId,
984     p.ProviderID
985   }).Distinct().OrderBy(sortvariable).ToList();
986
987 if (query.Count() != 0)
988 {
989   list<ProviderIRAScore> providerlist = new List<ProviderIRAScore>();
990
991   foreach (var q in query)
992   {
993     ProviderIRAScore provider = new ProviderIRAScore();
994
995     provider.ProviderID = q.ProviderID;
996     provider.ProviderName = q.ProviderName;
997     provider.ProviderScore = q.Projectedscore;
998     provider.ProviderScoreYear = q.ProjectedscoreYear;
999     provider.ProviderMidName = q.ProviderMidName;
1000    provider.ProviderLastName = q.ProviderLastName;
1001    provider.ProviderFirstName = q.ProviderFirstName;
1002    provider.HealthplanName = q.HealthplanName;
1003    provider.HealthplanId = q.HealthplanId;
1004    provider.ProjectedsMemberCount = q.ProjectedsMemberCount;
1005    provider.Projectedscore = q.Projectedscore;
1006    providerlist.Add(provider);
1007  }
1008
1009  return providerlist;
1010 }
```

A screenshot of the Microsoft Visual Studio IDE interface. The menu bar includes Edit, View, Project, Build, Debug, Team, Tools, Test, Analyze, Window, and Help. The toolbar shows icons for Undo, Redo, Cut, Copy, Paste, Find, Replace, and others. The solution explorer on the left lists 'MRAController.cs\*', 'MembersController.cs', 'Source Control Explorer', 'ACO\_MembersMart.cs', and 'Notification\_UserSpecific.cs'. The code editor window displays the 'UQP' class from 'UQP.Controllers.MRAController'. The code implements a provider list for a provider pie chart listing, handling pagination and sorting parameters. The code is heavily annotated with comments explaining its purpose and logic.

```
991 foreach (var q in query)
992 {
993     ProviderMRAScore provider = new ProviderMRAScore
994     {
995         FirstName = q.ProviderFirstName,
996         //MidName = q.ProviderMidName,
997         LastName = q.ProviderLastName,
998         healthplanname = q.HealthplanName,
999         Score = q.ProjectScore,
1000        membercount = q.ProjectMemberCount,
1001        ProviderId = q.ProviderID,
1002        healthplanid = q.HealthplanId
1003    };
1004    providerlist.Add(provider);
1005 }
1006 // Get's No of Rows Count
1007 if (member.report == false)
1008 {
1009     #region pagination
1010     int count = totalcount;
1011
1012     // Parameter is passed from Query string if it is null then it default Value will be pageNumber:1
1013     int CurrentPage = pagingParameterModel.pageNumber;
1014
1015     // Parameter is passed from Query string if it is null then it default Value will be pageSize:20
1016     int PageSize = pagingParameterModel.pageSize;
1017
1018     // Display TotalCount to Records to User
1019     int TotalCount = count;
1020
1021     // Calculating Totalpage by Dividing (No of Records / Pagesize)
1022     int TotalPages = (int) Math.Ceiling(count / (double)PageSize);
1023
1024     // Returns List of Customer after applying Paging
1025     var items = providerlist;//.Skip((CurrentPage - 1) * PageSize).Take(PageSize).ToList();
1026
1027     // If CurrentPage is greater than 1 means it has previousData
```

Microsoft Visual Studio

File View Project Build Debug Team Tools Test Analyze Window Help

Debug Any CPU IIS Express (Google Chrome)

MRAController.cs MembersController.cs ACO\_MembersMart.cs Notification\_UserSpecific.cs providerpiechartlisting(PagingParameterModel pagingParameterModel, I...

```
1085 // Get's No of Rows Count
1086 if (member.report == false)
1087 {
1088     #region pagination
1089     int count = totalcount;
1090
1091     // Parameter is passed from Query string if it is null then it default Value will be pageNumber:1
1092     int currentPage = pagingParameterModel.pageNumber;
1093
1094     // Parameter is passed from Query string if it is null then it default Value will be pageSize:20
1095     int pageSize = pagingParameterModel.pageSize;
1096
1097     // Display TotalCount to Records to User
1098     int totalCount = count;
1099
1100     // Calculating Totalpage by Dividing (No of Records / Pagesize)
1101     int totalPages = (int)Math.Ceiling(count / (double)pageSize);
1102
1103     // Returns List of Customer after applying Paging
1104     var items = providerlist;//.Skip((currentPage - 1) * pageSize).Take(pageSize).ToList();
1105
1106     // if currentPage is greater than 1 means it has previousPage
1107     var previousPage = currentPage > 1 ? "Yes" : "No";
1108
1109     // if TotalPages is greater than currentPage means it has nextPage
1110     var nextPage = currentPage < TotalPages ? "Yes" : "No";
1111
1112     // Object which we are going to send in header
1113     var paginationMetadata = new
1114     {
1115         totalCount = totalCount,
1116         pageSize = pageSize,
1117         currentPage = currentPage,
1118         totalPages = totalPages,
1119         previousPage,
1120         nextPage
1121     };
1122 }
```

Output Show output from: Source Control - Team Foundation

The screenshot shows a Microsoft Visual Studio interface with the following details:

- Menu Bar:** Edit, View, Project, Build, Debug, Team, Tools, Test, Analyze, Window, Help.
- Toolbox:** Standard icons for file operations like Open, Save, Print, etc.
- Toolbar:** Includes icons for Undo, Redo, Cut, Copy, Paste, Find, and others.
- Status Bar:** Shows "100%" at the bottom left.
- Code Editor:** Displays the `MRAController.cs` file content. The code is a C# class definition for `UQP.Controllers.MRAController`. It includes methods for generating reports and handling exceptions. The code is heavily annotated with comments and uses Newtonsoft.Json for JSON serialization.
- Tool Windows:** Source Control Explorer, ACO\_MembersMart.cs, and Notification\_UserSpecific.cs are visible on the right side.
- Task List:** Shows a providerpiechartlisting(PagingParameterModel pagingParameterModel, I...

```
UQP
1036     totalCount = totalCount,
1037     pageSize = PageSize,
1038     currentPage = CurrentPage,
1039     totalPages = TotalPages,
1040     previousPage,
1041     nextPage
1042     };
1043
1044     // Setting Header
1045
1046     System.Web.HttpContext.Current.Response.Headers.Add("Paging-Headers", Newtonsoft.Json.JsonConvert.SerializeObject(paginationMetadata));
1047     //HttpContext.Current.Response.Headers.Add("Filename", zipFileName);
1048     HttpContext.Current.Response.Headers.Add("Access-Control-Expose-Headers", "Paging-Headers");
1049     return Request.CreateResponse(HttpStatusCode.OK, items);
1050     #endregion
1051   }
1052   else
1053   {
1054     var rpobj = new ReportsController();
1055     {
1056       Request = new System.Net.Http.HttpRequestMessage(),
1057       Configuration = new HttpConfiguration()
1058     };
1059     return rpobj.GenerateNotifReport(null, null, null, null, null, null, "MRA Provider Listing", "excel", null, null, providerlist,
1060     null, null);
1061   }
1062 }
1063 return Request.CreateResponse(HttpStatusCode.OK, "Data not found");
1064 }
1065 }
1066 catch (Exception ex)
1067 {
1068   return Request.CreateResponse(HttpStatusCode.InternalServerError, ex.InnerException);
1069 }
1070 }
```

```
UQP
```

```
1059 }  
1060 }  
1061 }  
1062 }  
1063 }  
1064 }  
1065 }  
1066 }  
1067 }  
1068 }  
1069 }  
1070 }  
1071 }  
1072 }  
1073 }  
1074 }  
1075 [HttpPost]  
1076 [Route("api/mra/MRAReview/")]  
1077 public HttpResponseMessage MRAReview(MRAReviews hpMRA)  
1078 {  
1079     try  
1080     {  
1081         using (UQPEntities db = new UQPEntities(HttpContext.Current.Request.Headers.GetValues("IPA").FirstOrDefault()))  
1082         {  
1083             var splitted = hpMRA.createdbyfirstname.Split(new[] { ' ' }, 2);  
1084             string createdbyfirstname = splitted[0];  
1085             string createdbylastname = splitted[1];  
1086  
1087             string[] arr = hpMRA.conditions.Select(y => y.uniqueid).ToArray();  
1088             var checksid = db.MRAReviews.AsQueryable()  
1089             .Where(x => arr.Contains(x.uniqueid)).ToList();  
1090  
1091             string[] arr1 = checksid.Select(x => x.uniqueid).ToArray();  
1092  
1093             var newconditions = hpMRA.conditions  
1094             .Where(x => !arr1.Contains(x.uniqueid)).ToList();  
1095  
1096             //var checksid = db.MRAReviews.AsQueryable()  
1097             // .Where(x => x.subscriberid==hpMRA.subscriberid);  
1098             if (checksid.Count() != 0)  
1099             {  
1100                 foreach (var per in checksid)  
1101                 {  
1102                     foreach (var per2 in hpMRA.conditions.Where(x => x.id == per.id))  
1103                     {  
1104                         per.subscriberid = hpMRA.subscriberid;  
1105                         per.providerid = hpMRA.providerid;  
1106                     }  
1107                 }  
1108             }  
1109         }  
1110     }  
1111 }
```

The screenshot shows a Microsoft Visual Studio interface with the following details:

- Title Bar:** Debug - Any CPU - IIS Express (Google Chrome)
- Toolbars:** Standard, Windows Common, Windows Context
- Menus:** File, Edit, View, Insert, Tools, Options, Help
- Toolbox:** Standard
- Toolbars:** Standard, Windows Common, Windows Context
- Code Editor:** The main window displays the `MRAController.cs` file. The code is a C# class definition for `UQP.Controllers.MRAController`. It includes several properties and methods, such as `per`, `hpMRA`, and `per2`, which are likely used for mapping database records to objects. The code uses the `foreach` loop to iterate through a collection of `hpMRA` objects.
- Status Bar:** Shows "100 %".
- Output Window:** Shows "0 errors, 0 warnings".
- Task List:** Shows "0 items".
- Search Results:** Shows "0 items".
- Team Foundation:** Shows "0 items".

```
UQP.Controllers.MRAController
1101     foreach (var per2 in hpMRA.conversations.Where(x => x.id == per.id))
1102     {
1103         per.subscriberid = hpMRA.subscriberid;
1104         per.providerid = hpMRA.providerid;
1105         per.dos = per2.dos;
1106         per.diagnosicode = per2.diagnosicode;
1107         per.hcccondition = per2.hcccondition;
1108         per.hcccode = per2.hcccode;
1109         per.type = per2.type;
1110         per.lastofficevisit = hpMRA.lastofficevisit;
1111         per.nextofficevisit = hpMRA.nextofficevisit;
1112         per.followup = hpMRA.followup;
1113         per.subscriberfirstname = hpMRA.subscriberfirstname
1114         ;
1115         //subscribermidname = hpMRA.subscribermidname
1116         //;
1117         per.subscriberlastname = hpMRA.subscriberlastname
1118         ;
1119         per.providerfirstname = hpMRA.providerfirstname
1120         ;
1121         //providermidname = hpMRA.providermidname
1122         //;
1123         per.providerlastname = hpMRA.providerlastname
1124         ;
1125         per.createdbyfirstname = createdbyfirstname
1126         ;
1127         per.createdbymidname = ""
1128         ;
1129         per.createdbylastname = createdbylastname
1130         ;
1131         per.createdby = hpMRA.createdby
1132         ;
1133         per.createdDateByUQP = DateTime.Now;
1134         per.healthplanid = hpMRA.healthplanid;
1135         per.healthplanName = hpMRA.healthplanName;
1136         per.datenfhirth = hpMRA.datenfhirth;
1137         ;
1138     }
```

UQP - Microsoft Visual Studio

File Edit View Project Build Debug Team Tools Test Analyze Window Help

Debug Any CPU IIS Express (Google Chrome)

MRAController.cs MembersController.cs Source Control Explorer ACO\_MembersMart.cs Notification\_UserSpecific.cs

UQP

```
1138     per.dateofbirth = hpMRA.dateofbirth;
1139     per.gender = hpMRA.gender;
1140     per.Team = hpMRA.Team;
1141     per.TeamName = hpMRA.TeamName;
1142     per.HCCDescription = per2.HCCDescription;
1143     //id = per.id
1144
1145     }
1146 }
1147 }
1148 List<MRAREview> mrareviewlist = new List<MRAREview>();
1149 if (newconditions.Count() != 0)
{
1150
1151     foreach (var per in newconditions)
1152     {
1153         MRAREview mraobj = new MRAREview()
1154         {
1155             subscriberid = hpMRA.subscriberid,
1156             providerid = hpMRA.providerid,
1157             dos = per.dos,
1158             diagnosiscode = per.diagnosiscode,
1159             hcccondition = per.hcccondition,
1160             hccode = per.hccode,
1161             type = per.type,
1162             lastofficevisit = hpMRA.lastofficevisit,
1163             nextofficevisit = hpMRA.nextofficevisit,
1164             followup = hpMRA.followup,
1165             subscriberfirstname = hpMRA.subscriberfirstname
1166
1167             //subscribermidname = hpMRA.subscribermidname
1168             //,
1169             subscriberlastname = hpMRA.subscriberlastname
1170
1171
1172 }
```

```
1164 type = per.type,
1165 lastofficevisit = hpMRA.lastofficevisit,
1166 nextofficevisit = hpMRA.nextofficevisit,
1167 folowup = hpMRA.folowup,
1168 subscriberfirstname = hpMRA.subscriberfirstname
1169
1170 //subscribermidname = hpMRA.subscribermidname
1171 //,
1172 subscriberlastname = hpMRA.subscriberlastname
1173
1174 providerfirstname = hpMRA.providerfirstname
1175
1176 //providermidname = hpMRA.providermidname
1177 //,
1178 providerlastname = hpMRA.providerlastname
1179
1180 createdbyfirstname = createdbyfirstname
1181
1182 createdbymidname = ""
1183
1184 createdbylastname = createdbylastname
1185
1186 createdby = hpMRA.createdby
1187
1188 createdDateByUQP = DateTime.Now,
1189 healthplanid = hpMRA.healthplanid,
1190 healthplanName = hpMRA.healthplanName,
1191 dateofbirth = hpMRA.dateofbirth,
1192 gender = hpMRA.gender,
1193 Team = hpMRA.Team,
1194 TeamName = hpMRA.TeamName,
1195 HCCDescription = per.HCCDescription,
1196 id = per.id,
1197 uniqueid = per.uniqueid
1198
1199
1200
```

UQP - Microsoft Visual Studio

Edit View Project Build Debug Team Tools Test Analyze Window Help

192.61.99.225

MRAController.cs\* MembersController.cs Source Control Explorer ACO\_MembersMart.cs Notification\_UserSpecific.cs

```
1194
1195     team = hpMRA.team;
1196     TeamName = hpMRA.TeamName,
1197     HCCDescription = per.HCCDescription,
1198     id = per.id,
1199     uniqueid = per.uniqueid
1200
1201     };
1202     db.MRAReviews.Add(mraobj);
1203     mrareviewlist.Add(mraobj);
1204 }
1205
1206
1207     var membersmart = db.NewReportByMembers.AsQueryable().Where(x => x.SubscriberID == hpMRA.subscriberid);
1208     foreach (var per in membersmart)
1209     {
1210         per.isreviewed = true;
1211     }
1212     db.SaveChanges();
1213
1214     return Request.CreateResponse(HttpStatusCode.OK, mrareviewlist);
1215 }
1216
1217 }
1218 catch (Exception e)
1219 {
1220     return Request.CreateResponse(e.InnerException);
1221 }
1222 }
```

A screenshot of the Microsoft Visual Studio IDE. The main window shows a code editor with a large block of C# code. The code is annotated with line numbers from 1221 to 1256. It includes annotations like 'UQP' and 'References' pointing to specific lines. The code is part of a controller class, likely for an API endpoint. The top menu bar shows 'Debug' and 'Any CPU'. Below the menu, there are several tabs: 'MRAController.cs', 'MembersController.cs', 'Source Control Explorer', 'ACO\_MembersMart.cs', 'Notification\_UserSpecific.cs', and 'UQP.Controllers.MRAController'. The status bar at the bottom indicates '100 %'.

```
1221
1222 }
1223
1224
1225
1226
1227 [HttpPost]
1228 [Route("api/mra/MRAfollowup")]
1229 References
1230 public HttpResponseMessage MRAfollowup([FromUri]PagingParameterModel pagingParameterModel, InputDate inputDate)
1231 {
1232     try
1233     {
1234         using (UQPEntities db = new UQPEntities(HttpContext.Current.Request.Headers.GetValues("IPA").FirstOrDefault()))
1235         {
1236             int totalcount = 0;
1237             bool? auditstatus = null;
1238             if (inputDate.status != "tobefollowed" && inputDate.status != "Peer Review")
1239             {
1240                 auditstatus =
1241                     Convert.ToBoolean(inputDate.status);
1242             }
1243             List<MRAREview> reviewlist = new List<MRAREview>();
1244             var.query = (inputDate.report == false) ? (from mra in db.MRAREviews.AsQueryable()
1245                 Where(x =>
1246                     (inputDate.startdate.Year == 1 || (x.folowup >= System.Data.Entity.DbFunctions.TruncateTime(inputDate.startdate) &&
1247                         x.folowup <= System.Data.Entity.DbFunctions.TruncateTime(inputDate.enddate)))
1248                     && (inputDate.docsubmittedby == null || x.createdby == inputDate.docsubmittedby)
1249                     && (inputDate.healthplanid == 0 || x.healthplanid == inputDate.healthplanid)
1250                     && (inputDate.providerid == 0 || x.providerid == inputDate.providerid)
1251                     && (inputDate.condition == null || x.type == inputDate.condition)
1252                     && (inputDate.status == null ||
1253                         (inputDate.status == "tobefollowed" ? ((x.StatusReason == "Not Documented" || x.StatusReason == null) &&
1254                             (x.auditstatus != true || x.auditstatus == null)) : (inputDate.status == "Peer Review") ? x.UpperManagementFlag == true :
1255                             (x.auditstatus == auditstatus)))
1256                 )
1257                     select new
```



UQP

```
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304

    mra.createdbyfirstname,
    mra.createdbymidname,
    mra.createdbylastname,
    mra.createdDateByUQP,
    mra.dateofbirth,
    mra.gender,
    mra.followup

        }).Distinct()
    .OrderBy(x => new { x.providerlastname, x.providerfirstname, x.subscriberlastname, x.subscriberfirstname })
        .Skip(pagingParameterModel.pageSize * (pagingParameterModel.pageNumber - 1)).Take(pagingParameterModel.pageSize).ToList()
    :
    (from mra in db.MRAReviews.AsQueryable().
    Where(x =>
    (inputDate.startdate.Year == 1 || (x.followup >= System.Data.Entity.DbFunctions.TruncateTime(inputDate.startdate) && [
    x.followup <= System.Data.Entity.DbFunctions.TruncateTime(inputDate.enddate)])
    && (inputDate.docsubmittedby == null || x.createdby == inputDate.docsubmittedby)
    && (inputDate.healthplanid == 0 || x.healthplanid == inputDate.healthplanid)
    && (inputDate.providerid == 0 || x.providerid == inputDate.providerid)

    && (inputDate.condition == null || x.type == inputDate.condition)
    && (inputDate.status == null ||
    (inputDate.status == "tobefollowed" ?
    (x.StatusReason == "Not Documented" || x.StatusReason == null) &&
    (x.auditstatus != true || x.auditstatus == null)) : (inputDate.status == "Peer Review") ? x.UpperManagementFlag == true :
    (x.auditstatus == auditstatus)))
    )
        select new
    {
        mra.subscriberid,
        mra.subscriberfirstname,
```

```
Debug Any CPU IIS Express (Google Chrome) MRAController.cs MembersController.cs Source Control Explorer ACO_MembersMart.cs Notification_UserSpecific.cs MRAfollowup(PagingParameterModel pagingParameterModel, Imp
UQP UQP.Controllers.MRAController
1290     && (inputDate.docsubmittedby == null || x.createdby == inputDate.docsubmittedby)
1291     && (inputDate.healthplanid == 0 || x.healthplanid == inputDate.healthplanid)
1292     && (inputDate.providerid == 0 || x.providerid == inputDate.providerid)
1293
1294     && (inputDate.condition == null || x.type == inputDate.condition)
1295     && (inputDate.status == null ||
1296 (inputDate.status == "tobefollowed" ? (
1297     (x.StatusReason == "Not Documented" || x.StatusReason == null) &&
1298     (x.auditstatus != true || x.auditstatus == null)) : (inputDate.status == "Peer Review") ? x.UpperManagementFlag == true :
1299     (x.auditstatus == auditstatus)))
1300 )
1301         select new
1302     {
1303         mra.subscriberid,
1304         mra.subscriberfirstname,
1305         mra.subscribermidname,
1306         mra.subscriberlastname,
1307         mra.providerfirstname,
1308         mra.providermidname,
1309         mra.providerlastname,
1310         mra.healthplanid,
1311         mra.healthplanName,
1312         mra.providerid,
1313         mra.createdby,
1314         mra.createdbyfirstname,
1315         mra.createdbymidname,
1316         mra.createdbylastname,
1317         mra.createdDateByUQP,
1318         mra.dateofbirth,
1319         mra.gender,
1320         mra.followup
1321
1322
1323
1324
1325     }).Distinct()
100 % 4
Output
```

```
1323
1324
1325             }).Distinct()
1326             .OrderBy(x => new
1327             {
1328                 x.providerlastname,
1329                 x.providerfirstname,
1330                 x.subscriberlastname,
1331                 x.subscriberfirstname
1332             }).ToList();
1333             int totalcondition = 0;
1334             if (inputDate.report == false)
1335             {
1336                 var countvar = (from mra in db.MRAReviews.AsQueryable().
1337                     Where(x =>
1338                         (inputDate.startdate.Year == 1 || (x.followup >= System.Data.Entity.DbFunctions.TruncateTime(inputDate.startdate) &&
1339                             x.followup <= System.Data.Entity.DbFunctions.TruncateTime(inputDate.enddate)))
1340                             && (inputDate.docsubmittedby == null || x.createdby == inputDate.docsubmittedby)
1341                             && (inputDate.healthplanid == 0 || x.healthplanid == inputDate.healthplanid)
1342                             && (inputDate.providerid == 0 || x.providerid == inputDate.providerid)
1343
1344                             && (inputDate.condition == null || x.type == inputDate.condition)
1345                             && (inputDate.status == null ||
1346                                 (inputDate.status == "tobefollowed" ? (
1347                                     (x.StatusReason == "Not Documented" || x.StatusReason == null) &&
1348                                     (x.auditstatus != true || x.auditstatus == null)) : (inputDate.status == "Peer Review") ? x.UpperManagementFlag == true :
1349                                     (x.auditstatus == auditstatus)))
1350
1351             )
1352             select new
1353             {
1354                 mra.subscriberid,
1355                 mra.id
1356             });
1357
1358             totalcount = countvar.Select(x => x.subscriberid).Distinct().Count();
1359             totalcondition = countvar.Select(x => x.id).Count();
1360
1361         }
1362     }
1363 }
```

```
1359
1360
1361
1362     if (query.Count != 0)
1363     {
1364
1365         foreach (var q in query)
1366         {
1367             MRARReview reviewobj = new MRARReview
1368             {
1369                 id = totalcondition,
1370                 subscriberid = q.subscriberid,
1371                 providerid = q.providerid,
1372                 //                      type = q.type,
1373                 //dos = q.dos,
1374                 //diagnosiscode = q.diagnosiscode,
1375                 //hcccode = q.hcccode,
1376
1377                 // hcccondition = q.hcccondition,
1378                 // lastofficevisit = q.lastofficevisit,
1379                 // nextofficevisit = q.nextofficevisit,
1380                 // followup = q.followup,
1381                 subscriberfirstname = q.subscriberfirstname
1382
1383                 //subscribermidname = q.subscribermidname
1384                 //,
1385                 subscriberlastname = q.subscriberlastname
1386
1387                 providerfirstname = q.providerfirstname
1388
1389                 //providermidname = q.providermidname
1390                 //,
1391                 providerlastname = q.providerlastname
1392
1393                 healthplanName = q.healthplanName,
1394             };
1395         }
1396     }
1397 }
```



The screenshot shows a Microsoft Visual Studio interface with the following details:

- Open Files:** MRAController.cs, MembersController.cs, Source Control Explorer, ACO\_MembersMart.cs, Notification\_UserSpecific.cs.
- Current File:** UQP.UQP.Controllers.MRAController
- Code Content:** The code is a C# class definition for MRAController. It includes logic for handling pagination parameters from the query string, calculating total pages, and creating a pagination metadata object. The code is heavily annotated with comments explaining its purpose.

```
1413     if (inputDate.report == false)
1414     {
1415         #region pagination
1416         int count = totalcount;
1417
1418         // Parameter is passed from Query string if it is null then it default Value will be pageNumber:1
1419         int CurrentPage = pagingParameterModel.pageNumber;
1420
1421         // Parameter is passed from Query string if it is null then it default Value will be pageSize:20
1422         int PageSize = pagingParameterModel.pageSize;
1423
1424         // Display TotalCount to Records to User
1425         int TotalCount = count;
1426
1427         // Calculating Totalpage by Dividing (No of Records / Pagesize)
1428         int TotalPages = (int)Math.Ceiling(count / (double)PageSize);
1429
1430         // Returns List of Customer after applying Paging
1431         var items = reviewlist;//.Skip((CurrentPage - 1) * PageSize).Take(PageSize).ToList();
1432
1433         // if CurrentPage is greater than 1 means it has previousPage
1434         var previousPage = CurrentPage > 1 ? "Yes" : "No";
1435
1436         // if TotalPages is greater than CurrentPage means it has nextPage
1437         var nextPage = CurrentPage < TotalPages ? "Yes" : "No";
1438
1439         // Object which we are going to send in header
1440         var paginationMetadata = new
1441         {
1442             totalCount = TotalCount,
1443             pageSize = PageSize,
1444             currentPage = CurrentPage,
1445             totalPages = TotalPages,
1446             previousPage,
1447             nextPage
1448         };
1449     }
1450 }
```

The screenshot shows a Microsoft Visual Studio interface with the following details:

- Open Files:** MRAController.cs\*, MembersController.cs, Source Control Explorer, ACO\_MembersMart.cs, Notification\_UserSpecific.cs.
- Current File:** UQP.Controllers.MRAController
- Code Preview:** The code is displayed in a large central pane, showing a series of numbered lines (1431 to 1466) of C# code related to pagination and reporting.
- Toolbars and Menus:** Standard Visual Studio toolbars and menus are visible along the top.
- Status Bar:** The status bar at the bottom shows "Output" and "100 %".

```
1431 var items = reviewlist;//.Skip((CurrentPage - 1) * PageSize).Take(PageSize).ToList();
1432 
1433 // if CurrentPage is greater than 1 means it has previousPage
1434 var previousPage = CurrentPage > 1 ? "Yes" : "No";
1435 
1436 // if TotalPages is greater than CurrentPage means it has nextPage
1437 var nextPage = CurrentPage < TotalPages ? "Yes" : "No";
1438 
1439 // Object which we are going to send in header
1440 var paginationMetadata = new
1441 {
1442     totalCount = TotalCount,
1443     pageSize = PageSize,
1444     currentPage = CurrentPage,
1445     totalPages = TotalPages,
1446     previousPage,
1447     nextPage
1448 };
1449 
1450 // Setting Header
1451 .System.Web.HttpContext.Current.Response.Headers.Add("Paging-Headers", Newtonsoft.Json.JsonConvert.SerializeObject(paginationMetadata));
1452 //HttpContext.Current.Response.Headers.Add("Filename", zipFileName);
1453 HttpContext.Current.Response.Headers.Add("Access-Control-Expose-Headers", "Paging-Headers");
1454 return Request.CreateResponse(HttpStatusCode.OK, items);
1455 #endregion
1456 }
1457 else
1458 {
1459     var rpoj = new ReportsController
1460     {
1461         Request = new System.Net.Http.HttpRequestMessage(),
1462         Configuration = new HttpConfiguration()
1463     };
1464     return rpoj.GenerateNotifReport(null, null, null, null, null, null, "Follow Up Report", "excel", null, null, null, null, reviewlist, null);
1465 }
1466 }
```

```
1440     previousPage,
1441     nextPage
1442 };
1443
1444 // Setting Header
1445
1446 System.Web.HttpContext.Current.Response.Headers.Add("Paging-Headers", Newtonsoft.Json.JsonConvert.SerializeObject(paginationMetadata));
1447 //HttpContext.Current.Response.Headers.Add("Filename", zipFileName);
1448 HttpContext.Current.Response.Headers.Add("Access-Control-Expose-Headers", "Paging-Headers");
1449 return Request.CreateResponse(HttpStatusCode.OK, items);
1450 #endregion
1451 }
1452 else
1453 {
1454     var rpoj = new ReportsController
1455     {
1456         Request = new System.Net.Http.HttpRequestMessage(),
1457         Configuration = new HttpConfiguration()
1458     };
1459     rpoj.GenerateNotifReport(null, null, null, null, null, null, "Follow Up Report", "excel", null, null, null, null, null, reviewlist, null, null);
1460 }
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481 }
```

Output  
Show output from: Source Control - Team Foundation

Ln 1466 Col 29 Ch 29

```
1477
1478
1479
1480
1481
1482
1483
1484 //     [HttpPost]
1485 //     [Route("api/mra/MRASuspected")]
1486 //     public HttpResponseMessage MRASuspected(InputDate inputDate)
1487 //     {
1488
1489 //         try
1490 //         {
1491 //             using (UQPEntities db = new UQPEntities(HttpContext.Current.Request.Headers.GetValues("IPA").FirstOrDefault()))
1492 //             {
1493 //                 bool? auditstatus = null;
1494 //                 if (inputDate.status != "tobefollowed")
1495 //                 {
1496 //                     auditstatus =
1497 //                         Convert.ToBoolean(inputDate.status);
1498 //                 }
1499
1500
1501
1502
1503
1504 //                 var condition = db.MRAReviews.Where(x => x.subscriberid == inputDate.subscriberid &&
1505 //                 (inputDate.startdate.Year == 1 ||
1506 //                  (x.followup >= inputDate.startdate && x.followup <= inputDate.enddate))
1507 //                  && (inputDate.docsubmittedby == null || x.createdby == inputDate.docsubmittedby)
1508 //                  && (inputDate.healthplanid == 0 || x.healthplanid == inputDate.healthplanid)
1509 //                  && (inputDate.providerid == 0 || x.providerid == inputDate.providerid)
1510
1511 //                  && (inputDate.condition == null || x.type == inputDate.condition)
```

100 %

Output

Show output from: Source Control - Team Foundation

Error List Output

Ln 1466

Col 29

Ch 29

Ready

Scanned with CamScanner

```
1509 // && (inputDate.providerid == 0 || x.providerid == inputDate.providerid)
1510 //
1511 // && (inputDate.condition == null || x.type == inputDate.condition)
1512 // && (inputDate.status == null ||
1513 // (inputDate.status == "tobefollowed" ? (
1514 // (x.StatusReason == "Not Documented" || x.StatusReason == null) &&
1515 // (x.auditstatus != true || x.auditstatus == null)) :
1516 // x.auditstatus == auditstatus));
1517 //
1518 // var doc =
1519 // db.MRAReviewDocuments.Where(x => x.SubscriberId == inputDate.subscriberid).ToList();
1520 //
1521 // MRAReviews mraobj = new MRAReviews();
1522 // List<MRAReviews> conditionlist = new List<MRAReviews>();
1523 // mraobj.lastofficevisit = condition.FirstOrDefault().lastofficevisit;
1524 // mraobj.nextofficevisit = condition.FirstOrDefault().nextofficevisit;
1525 // mraobj.followup = condition.FirstOrDefault().followup;
1526 // mraobj.subscriberfirstname = condition.FirstOrDefault().subscriberfirstname
1527 //
1528 // mraobj.subscribermidname = condition.FirstOrDefault().subscribermidname
1529 //
1530 // mraobj.subscriberlastname = condition.FirstOrDefault().subscriberlastname
1531 //
1532 // mraobj.providerfirstname = condition.FirstOrDefault().providerfirstname
1533 //
1534 // mraobj.providermidname = condition.FirstOrDefault().providermidname
1535 //
1536 // mraobj.providerlastname = condition.FirstOrDefault().providerlastname
1537 //
1538 // mraobj.createdbyfirstname = condition.FirstOrDefault().createdbyfirstname
1539 //
1540 // mraobj.createdbymidname = ""
1541 //
1542 // mraobj.createdbylastname = condition.FirstOrDefault().createdbylastname
1543 //
1544 //
```

Scanned with CamScanner

```
1539 // mraobj.createdbyfirstname = condition.FirstOrDefault().createdbyfirstname
1540 //;
1541 // mraobj.createdbymidname = ""
1542 //;
1543 // mraobj.createdbylastname = condition.FirstOrDefault().createdbylastname
1544 //;
1545 // mraobj.createdby = condition.FirstOrDefault().createdby
1546 //;
1547 // mraobj.createdDateByUQP = condition.FirstOrDefault().createdDateByUQP;
1548 // mraobj.healthplanid = condition.FirstOrDefault().healthplanid;
1549 // mraobj.healthplanName = condition.FirstOrDefault().healthplanName;
1550 // mraobj.subscriberid = condition.FirstOrDefault().subscriberid;
1551 // mraobj.providerid = condition.FirstOrDefault().providerid;
1552
1553
1554
1555
1556
1557
1558 // foreach (var per in condition.ToList())
1559 //{
1560 //     MRAReviews mracondition = new MRAReviews();
1561
1562
1563
1564 //     mracondition.id = per.id;
1565 //     mracondition.dos = per.dos;
1566 //     mracondition.diagnosiscode = per.diagnosiscode;
1567 //     mracondition.hcccondition = per.hcccondition;
1568 //     mracondition.hcccode = per.hcccode;
1569 //     mracondition.type = per.type;
1570 //     mracondition.auditstatus = per.auditstatus;
1571 //     mracondition.auditorfirstname = per.auditorfirstname;
1572 //     mracondition.auditorlastname = per.auditorlastname;
1573 //     mracondition.auditedDate = per.auditedDate;
1574 // }
```

```
1563 // 
1564 // 
1565 // mracondition.id = per.id;
1566 // mracondition.dos = per.dos;
1567 // mracondition.diagnosiscode = per.diagnosiscode;
1568 // mracondition.hcccondition = per.hcccondition;
1569 // mracondition.hcccode = per.hcccode;
1570 // mracondition.type = per.type;
1571 // mracondition.auditstatus = per.auditstatus;
1572 // mracondition.auditorfirstname = per.auditorfirstname;
1573 // mracondition.auditorlastname = per.auditorlastname;
1574 // mracondition.audittedDate = per.audittedDate;
1575 // mracondition.auditedby = per.auditedby;
1576 // mracondition.StatusReason = per.StatusReason;
1577 // mracondition.HCCDescription = per.HCCDescription;
1578 // 
1579 // List<MRAReviewDocument> documentlist = new List<MRAReviewDocument>();
1580 // foreach (var per2 in doc)
1581 // {
1582 //     if (per2.MRAID == per.id)
1583 //     {
1584 //         MRAReviewDocument documentobj = new MRAReviewDocument();
1585 // 
1586 //         documentobj.SubscriberId = per2.SubscriberId;
1587 //         documentobj.Filename = per2.Filename;
1588 //         documentobj.Filepath = per2.Filepath + per2.Filename;
1589 //         documentobj.MRAID = per2.MRAID;
1590 // 
1591 //         documentlist.Add(documentobj);
1592 //     }
1593 // }
1594 // 
1595 // mracondition.documents = documentlist;
1596 // 
1597 // conditionlist.Add(mracondition);
1598 // 
```

The screenshot shows a Microsoft Visual Studio interface with the following details:

- Title Bar:** Shows "MRAController.cs" as the active file, "Debug" mode, "Any CPU" architecture, and "IIS Express (Google Chrome)" as the running application.
- Toolbars:** Standard Visual Studio toolbars for Solution Explorer, Toolbox, and other development tools.
- Code Editor:** The main window displays C# code for the `MRAController.cs` file. The code is heavily annotated with numerous double slashes (//) preceding every line of code, indicating it is a comment. The code itself appears to be part of a larger function, likely handling document or review processing.
- Status Bar:** Shows "100 % 4" at the bottom left and "Output" at the bottom right.

```
1590 // documentlist.Add(documentobj);
1591 // }
1592 // }
1593 // mracondition.documents = documentlist;
1594 //
1595 // conditionlist.Add(mracondition);
1596 //
1597 // }
1598 // mraobj.conditions = conditionlist;
1599 // List<MRAREviewDocument> documentlist = new List<MRAREviewDocument>();
1600 // foreach (var per2 in doc)
1601 // {
1602 //     MRAREviewDocument documentobj = new MRAREviewDocument
1603 //     {
1604 //         SubscriberId = per2.SubscriberId,
1605 //         Filename = per2.Filename,
1606 //         Filepath = per2.Filepath + per2.Filename
1607 //     };
1608 //     documentlist.Add(documentobj);
1609 // }
1610 // mraobj.documents = documentlist;
1611 //
1612 // }
1613 // return Request.CreateResponse(HttpStatusCode.OK, conditionlist);
1614 //
1615 // }
1616 //
1617 //
1618 //
1619 //
1620 //
1621 //
1622 // }
1623 //
1624 // }
1625 //
```

UQP - Microsoft Visual Studio

File Edit View Project Build Debug Team Tools Test Analyze Window Help

Debug Any CPU IIS Express (Google Chrome)

MRAController.cs\* MembersController.cs Source Control Explorer ACO\_MembersMart.cs Notification\_UserSpecific.cs MRAfollowup(PagingParameterModel pagingParameterModel)

UQP

```
1611 //      documentlist.Add(documentobj);
1612 //      //}
1613 //      mraobj.documents = documentlist;
1614
1615
1616
1617
1618
1619
1620
1621
1622 //      return Request.CreateResponse(HttpStatusCode.OK, conditionlist);
1623 //    }
1624
1625 //  }
1626 //  catch (Exception e)
1627 //  {
1628 //    return Request.CreateResponse(e.InnerException);
1629 //  }
1630 //}
1631
1632
1633
1634
1635
```

The screenshot shows a Microsoft Visual Studio interface with the following details:

- Project Explorer**: Shows a tree view of project files.
- Toolbox**: Standard .NET development tools.
- Code Editor**: The main window displays the `MRAController.cs` file. The code is a C# class definition for `UQP.Controllers.MRAController`. It includes annotations for `[HttpPost]` and `[Route("api/mra/MRASuspected")]`. The class contains a method `MRASuspected` that takes an `InputDate` parameter and returns an `HttpResponseMessage`. The method uses a `UQPEntities` database context to query `MRAReviews` based on various conditions like year, follow-up date, and provider status. The code is heavily annotated with line numbers from 1635 to 1669.
- Task List**: Shows a list of tasks or errors.
- Solution Explorer**: Shows the project structure.
- Source Control Explorer**: Shows version control information.
- ACO\_MembersMart.cs**: Another code file is visible in the background.
- Notification\_UserSpecific.cs**: Another code file is visible in the background.
- Output**: Shows build logs.

```
1635  
1636  
1637  
1638     [HttpPost]  
1639     [Route("api/mra/MRASuspected")]  
1640     1reference  
1641     public HttpResponseMessage MRASuspected(InputDate inputDate)  
1642     {  
1643         try  
1644         {  
1645             using (UQPEntities db = new UQPEntities(HttpContext.Current.Request.Headers.GetValues("IPA").FirstOrDefault()))  
1646             {  
1647                 bool? auditstatus = null;  
1648                 if (inputDate.status != "tobefollowed")  
1649                 {  
1650                     auditstatus =  
1651                         Convert.ToBoolean(inputDate.status);  
1652                 }  
1653  
1654  
1655  
1656  
1657                 var condition = db.MRAReviews.Where(x => x.subscriberid == inputDate.subscriberid &&  
1658                 (inputDate.startdate.Year == 1 ||  
1659                     (x.followup >= inputDate.startdate && x.followup <= inputDate.enddate))  
1660                     && (inputDate.docsubmittedby == null || x.createdby == inputDate.docsubmittedby)  
1661                     && (inputDate.healthplanid == 0 || x.healthplanid == inputDate.healthplanid)  
1662                     && (inputDate.providerid == 0 || x.providerid == inputDate.providerid)  
1663  
1664                     && (inputDate.condition == null || x.type == inputDate.condition)  
1665                     && (inputDate.status == null ||  
1666                     (inputDate.status == "tobefollowed" ?  
1667                         (x.StatusReason == "Not Documented" || x.StatusReason == null) &&  
1668                         (x.auditstatus != true || x.auditstatus == null)) :  
1669                         x.auditstatus == auditstatus));  
100 % 4
```

The screenshot shows a Microsoft Visual Studio interface with the following details:

- File Explorer**: Shows files like MRAController.cs, MembersController.cs, Source Control Explorer, ACO\_MembersMart.cs, Notification\_UserSpecific.cs, and UOP.
- MRAController.cs\*** is the active file, displaying C# code for a controller. The code is heavily annotated with line numbers (1661 to 1696) and conditionals for providerid, status, and auditstatus.
- Output** tab: Shows "Show output from: Source Control - Team Foundation".

```
1661     && (inputDate.providerid == 0 || x.providerid == inputDate.providerid)
1662
1663     && (inputDate.condition == null || x.type == inputDate.condition)
1664     && (inputDate.status == null ||
1665         (inputDate.status == "tobefollowed" ? (
1666             (x.StatusReason == "Not Documented" || x.StatusReason == null) &&
1667             (x.auditstatus != true || x.auditstatus == null)) :
1668             x.auditstatus == auditstatus)));
1669
1670     var doc = db.MRAReviewDocuments.Where(x => x.SubscriberId == inputDate.subscriberid &&
1671     x.status == true).ToList();
1672
1673     MRAReviews mraobj = new MRAReviews();
1674     List<MRAReviewConditions> conditionlist = new List<MRAReviewConditions>();
1675     mraobj.lastofficevisit = condition.FirstOrDefault().lastofficevisit;
1676     mraobj.nextofficevisit = condition.FirstOrDefault().nextofficevisit;
1677     mraobj.followup = condition.FirstOrDefault().followup;
1678     mraobj.subscriberfirstname = condition.FirstOrDefault().subscriberfirstname
1679     ;
1680     mraobj.subscribermidname = condition.FirstOrDefault().subscribermidname
1681     ;
1682     mraobj.subscriberlastname = condition.FirstOrDefault().subscriberlastname
1683     ;
1684     mraobj.providerfirstname = condition.FirstOrDefault().providerfirstname
1685     ;
1686     mraobj.providermidname = condition.FirstOrDefault().providermidname
1687     ;
1688     mraobj.providerlastname = condition.FirstOrDefault().providerlastname
1689     ;
1690     mraobj.createdbyfirstname = condition.FirstOrDefault().createdbyfirstname
1691     ;
1692     mraobj.createdbymidname = "";
1693     ;
1694     mraobj.createdbylastname = condition.FirstOrDefault().createdbylastname
1695     ;
1696     ;
```

The screenshot shows a Microsoft Visual Studio interface with the following details:

- Title Bar:** Shows "Debug Any CPU IIS Express (Google Chrome)"
- Toolbars:** "Server Explorer" and "Toolbox" are visible on the left.
- Code Editor:** The main window displays the "MRAController.cs" file. The code is a C# class definition for "UQP.Controllers.MRAController". The code is heavily annotated with line numbers from 1685 to 1720. It includes assignments to properties like providermidname, providerlastname, createdbyfirstname, createdbymidname, createdbylastname, createdby, createdDateByUQP, healthplanid, healthplanName, subscriberid, providerid, and a foreach loop that initializes an object named "mracondition".
- Status Bar:** Shows "Output" and "100 %".

```
1685 ; mraobj.providermidname = condition.FirstOrDefault().providermidname
1686 ;
1687 ; mraobj.providerlastname = condition.FirstOrDefault().providerlastname
1688 ;
1689 ; mraobj.createdbyfirstname = condition.FirstOrDefault().createdbyfirstname
1690 ;
1691 ; mraobj.createdbymidname = ""
1692 ;
1693 ; mraobj.createdbylastname = condition.FirstOrDefault().createdbylastname
1694 ;
1695 ; mraobj.createdby = condition.FirstOrDefault().createdby
1696 ;
1697 ;
1698 ; mraobj.createdDateByUQP = condition.FirstOrDefault().createdDateByUQP;
1699 mraobj.healthplanid = condition.FirstOrDefault().healthplanid;
1700 mraobj.healthplanName = condition.FirstOrDefault().healthplanName;
1701 mraobj.subscriberid = condition.FirstOrDefault().subscriberid;
1702 mraobj.providerid = condition.FirstOrDefault().providerid;
1703
1704
1705
1706
1707
1708
1709
1710 foreach (var per in condition.ToList())
1711 {
1712     MRAReviewConditions mracondition = new MRAReviewConditions();
1713
1714
1715
1716     mracondition.id = per.id;
1717     mracondition.uniqueid = per.uniqueid;
1718     mracondition.dos = per.dos;
1719     mracondition.diagnosiscode = per.diagnosiscode;
1720 }
```

```
1712     MRAReviewConditions mracondition = new MRAReviewConditions();
1713
1714
1715
1716
1717     mracondition.id = per.id;
1718     mracondition.uniqueid = per.uniqueid;
1719     mracondition.dos = per.dos;
1720     mracondition.diagnosiscode = per.diagnosiscode;
1721     mracondition.hcccondition = per.hcccondition;
1722     mracondition.hcccode = per.hcccode;
1723     mracondition.type = per.type;
1724     mracondition.auditstatus = per.auditstatus;
1725     mracondition.auditorfirstname = per.auditorfirstname;
1726     mracondition.auditorlastname = per.auditorlastname;
1727     mracondition.auditedDate = per.auditedDate;
1728     mracondition.auditedby = per.auditedby;
1729     mracondition.StatusReason = per.StatusReason;
1730     mracondition.HCCDescription = per.HCCDescription;
1731     List<MRAReviewDocument> documentlist = new List<MRAReviewDocument>();
1732     foreach (var per2 in doc)
1733     {
1734         if (per2.MRAId == per.uniqueid)
1735         {
1736             MRAReviewDocument documentobj = new MRAReviewDocument();
1737
1738             documentobj.SubscriberId = per2.SubscriberId;
1739             documentobj.Filename = per2.Filename;
1740             documentobj.Filepath = per2.Filepath + per2.Filename;
1741             documentobj.MRAId = per2.MRAId;
1742
1743
1744             documentlist.Add(documentobj);
1745         }
1746     }
1747 }
```

```
1739  
1740  
1741  
1742  
1743  
1744  
1745  
1746  
1747  
1748  
1749  
1750  
1751  
1752  
1753  
1754  
1755  
1756  
1757  
1758  
1759  
1760  
1761  
1762  
1763  
1764  
1765        return Request.CreateResponse(HttpStatusCode.OK, mraobj);  
1766 }  
1767 }  
1768 }  
1769 catch (Exception e)  
1770 {  
1771     return Request.CreateResponse(e.InnerException);  
1772 }  
1773 }  
1774 }
```

Output

Show output from: Source Control - Team Foundation

```
UQP UQP.Controllers.MRAController MRAfollowup(PagingParameterModel page)
1775
1776
1777
1778
1779     [HttpGet]
1780     [Route("api/MRA/removefile/{uniqueid}/{filename}")]
1781     public HttpResponseMessage DeleteFile(string uniqueid, string filename)
1782     {
1783         try
1784         {
1785             using (UQPEntities db = new UQPEntities(HttpContext.Current.Request.Headers.GetValues("IPA").FirstOrDefault()))
1786             {
1787                 MRAReviewDocument objsub = (from a in db.MRAReviewDocuments
1788                                              where a.MRAId == uniqueid && a.Filename == filename
1789                                              select a).FirstOrDefault();
1790
1791                 if (objsub == null)
1792                 {
1793                     return Request.CreateResponse(HttpStatusCode.NotFound, "Data not found");
1794                 }
1795                 else
1796                 {
1797                     db.MRAReviewDocuments.Remove(objsub);
1798                     db.SaveChanges();
1799
1800                     InputDate objinp = new InputDate();
1801                     objinp.subscriberid = objsub.SubscriberId;
1802                     objinp.providerid = 0;
1803                     objinp.healthplanid = 0;
1804
1805
1806
1807
1808
1809             return MRASuspected(objinp);
1810         }
1811     }
1812 }
```

Microsoft Visual Studio 16.8.1.09215

File Edit View Project Build Debug Team Tools Test Analyze Window Help

Debug Any CPU IIS Express (Google Chrome)

MRAController.cs MembersController.cs Source Control Explorer ACO\_MembersMart.cs Notification\_UserSpecific.cs MRAfollowup(PagingParameterModel.cs)

UQP

```
1792     .return Request.CreateResponse(HttpStatusCode.NotFound, "Data not found");
1793 }
1794 else
1795 {
1796
1797     db.MRAReviewDocuments.Remove(objsub);
1798     db.SaveChanges();
1799
1800     InputDate objinp = new InputDate();
1801     objinp.subscriberid = objsub.SubscriberId;
1802     objinp.providerid = 0;
1803     objinp.healthplanid = 0;
1804
1805
1806
1807
1808     return MRASuspected(objinp);
1809
1810
1811
1812
1813     }
1814 }
1815 catch (Exception ex)
1816 {
1817     return Request.CreateResponse(HttpStatusCode.InternalServerError, ex.Message);
1818 }
1819
1820
1821
1822 }
```

Microsoft Visual Studio

File Edit View Project Build Debug Team Tools Test Analyze Window Help

UQP

MRAController.cs\* MembersController.cs Source Control Explorer ACO\_MembersMart.cs Notification\_UserSpecific.cs MRAfollowup(PagingParameterModel pagingParameterModel, InputData

```
1825 [HttpPost]
1826 [Route("api/mra/prevelance")]
1827 public HttpResponseMessage prevelance(InputDate inputDate)
1828 {
1829     try
1830     {
1831         using (UQPEntities db = new UQPEntities(HttpContext.Current.Request.Headers.GetValues("IPA").FirstOrDefault()))
1832         {
1833             var result = db.ProviderPrevelanceReport(inputDate.year,(int?)inputDate.healthplanid).ToList();
1834             // int totalcount = result.Where(x => x.provider_id == inputDate.providerid && (inputDate.healthplanid==0 || x.HealthPlanID==
1835             // inputDate.healthplanid)).Count();
1836
1837             var membersresult = db.MRA_ReportByProviderInsurance.Where(x=>x.ProjectectedScoreYear==inputDate.year).ToList();
1838
1839             var providerpercent = result.Where(x => x.provider_id == inputDate.providerid
1840             && (inputDate.healthplanid == 0 || x.HealthPlanID ==
1841             inputDate.healthplanid));
1842             //OrderBy(x => x.value)
1843             //Skip(inputDate.pageSize * (inputDate.pageNumber - 1)).Take(inputDate.pageSize).ToList();
1844             var overallpercent = result.Where(x => x.HealthPlanID == inputDate.healthplanid);
1845             var overallprovidermember = membersresult.Where(x => (inputDate.healthplanid == 0 || x.HealthPlanID ==
1846             inputDate.healthplanid) && x.ProviderID==inputDate.providerid
1847             && x.ProjectectedScoreYear==inputDate.year)
1848             .Sum(x=>x.ProjectectedMemberCount);
1849             double? overallmember = membersresult.Where(x => (inputDate.healthplanid == 0 || x.HealthPlanID ==
1850             inputDate.healthplanid)
1851             && x.ProjectectedScoreYear == inputDate.year)
1852             .Sum(x => x.ProjectectedMemberCount);
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2498
2499
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2598
2599
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2619
2620
2621
2622
2623
2624
2625
2626
2627
2628
2629
2629
2630
2631
2632
2633
2634
2635
2636
2637
2638
2639
2639
2640
2641
2642
2643
2644
2645
2646
2647
2648
2649
2649
2650
2651
2652
2653
2654
2655
2656
2657
2658
2659
2659
2660
2661
2662
2663
2664
2665
2666
2667
2668
2669
2669
2670
2671
2672
2673
2674
2675
2676
2677
2678
2679
2679
2680
2681
2682
2683
2684
2685
2686
2687
2688
2689
2689
2690
2691
2692
2693
2694
2695
2696
2697
2698
2698
2699
2699
2700
2701
2702
2703
2704
2705
2706
2707
2708
2709
2709
2710
2711
2712
2713
2714
2715
2716
2717
2718
2719
2719
2720
2721
2722
2723
2724
2725
2726
2727
2728
2729
2729
2730
2731
2732
2733
2734
2735
2736
2737
2738
2739
2739
2740
2741
2742
2743
2744
2745
2746
2747
2748
2749
2749
2750
2751
2752
2753
2754
2755
2756
2757
2758
2759
2759
2760
2761
2762
2763
2764
2765
2766
2767
2768
2769
2769
2770
2771
2772
2773
2774
2775
2776
2777
2778
2779
2779
2780
2781
2782
2783
2784
2785
2786
2787
2788
2789
2789
2790
2791
2792
2793
2794
2795
2796
2797
2798
2798
2799
2799
2800
2801
2802
2803
2804
2805
2806
2807
2808
2809
2809
2810
2811
2812
2813
2814
2815
2816
2817
2818
2819
2819
2820
2821
2822
2823
2824
2825
2826
2827
2828
2829
2829
2830
2831
2832
2833
2834
2835
2836
2837
2838
2839
2839
2840
2841
2842
2843
2844
2845
2846
2847
2848
2849
2849
2850
2851
2852
2853
2854
2855
2856
2857
2858
2859
2859
2860
2861
2862
2863
2864
2865
2866
2867
2868
2869
2869
2870
2871
2872
2873
2874
2875
2876
2877
2878
2879
2879
2880
2881
2882
2883
2884
2885
2886
2887
2888
2889
2889
2890
2891
2892
2893
2894
2895
2896
2897
2898
2898
2899
2899
2900
2901
2902
2903
2904
2905
2906
2907
2908
2909
2909
2910
2911
2912
2913
2914
2915
2916
2917
2918
2919
2919
2920
2921
2922
2923
2924
2925
2926
2927
2928
2929
2929
2930
2931
2932
2933
2934
2935
2936
2937
2938
2939
2939
2940
2941
2942
2943
2944
2945
2946
2947
2948
2949
2949
2950
2951
2952
2953
2954
2955
2956
2957
2958
2959
2959
2960
2961
2962
2963
2964
2965
2966
2967
2968
2969
2969
2970
2971
2972
2973
2974
2975
2976
2977
2978
2979
2979
2980
2981
2982
2983
2984
2985
2986
2987
2988
2989
2989
2990
2991
2992
2993
2994
2995
2996
2997
2998
2998
2999
2999
3000
3001
3002
3003
3004
3005
3006
3007
3008
3009
3009
3010
3011
3012
3013
3014
3015
3016
3017
3018
3019
3019
3020
3021
3022
3023
3024
3025
3026
3027
3028
3029
3029
3030
3031
3032
3033
3034
3035
3036
3037
3038
3039
3039
3040
3041
3042
3043
3044
3045
3046
3047
3048
3049
3049
3050
3051
3052
3053
3054
3055
3056
3057
3058
3059
3059
3060
3061
3062
3063
3064
3065
3066
3067
3068
3069
3069
3070
3071
3072
3073
3074
3075
3076
3077
3078
3079
3079
3080
3081
3082
3083
3084
3085
3086
3087
3088
3089
3089
3090
3091
3092
3093
3094
3095
3096
3097
3098
3098
3099
3099
3100
3101
3102
3103
3104
3105
3106
3107
3108
3109
3109
3110
3111
3112
3113
3114
3115
3116
3117
3118
3119
3119
3120
3121
3122
3123
3124
3125
3126
3127
3128
3129
3129
3130
3131
3132
3133
3134
3135
3136
3137
3138
3139
3139
3140
3141
3142
3143
3144
3145
3146
3147
3148
3149
3149
3150
3151
3152
3153
3154
3155
3156
3157
3158
3159
3159
3160
3161
3162
3163
3164
3165
3166
3167
3168
3169
3169
3170
3171
3172
3173
3174
3175
3176
3177
3178
3179
3179
3180
3181
3182
3183
3184
3185
3186
3187
3188
3189
3189
3190
3191
3192
3193
3194
3195
3196
3197
3198
3198
3199
3199
3200
3201
3202
3203
3204
3205
3206
3207
3208
3209
3209
3210
3211
3212
3213
3214
3215
3216
3217
3218
3219
3219
3220
3221
3222
3223
3224
3225
3226
3227
3228
3229
3229
3230
3231
3232
3233
3234
3235
3236
3237
3238
3239
3239
3240
3241
3242
3243
3244
3245
3246
3247
3248
3249
3249
3250
3251
3252
3253
3254
3255
3256
3257
3258
3259
3259
3260
3261
3262
3263
3264
3265
3266
3267
3268
3269
3269
3270
3271
3272
3273
3274
3275
3276
3277
3278
3279
3279
3280
3281
3282
3283
3284
3285
3286
3287
3288
3289
3289
3290
3291
3292
3293
3294
3295
3296
3297
3298
3298
3299
3299
3300
3301
3302
3303
3304
3305
3306
3307
3308
3309
3309
3310
3311
3312
3313
3314
3315
3316
3317
3318
3319
3319
3320
3321
3322
3323
3324
3325
3326
3327
3328
3329
3329
3330
3331
3332
3333
3334
3335
3336
3337
3338
3339
3339
3340
3341
3342
3343
3344
3345
3346
3347
3348
3349
3349
3350
3351
3352
3353
3354
3355
3356
3357
3358
3359
3359
3360
3361
3362
3363
3364
3365
3366
3367
3368
3369
3369
3370
3371
3372
3373
3374
3375
3376
3377
3378
3379
3379
3380
3381
3382
3383
3384
3385
3386
3387
3388
3389
3389
3390
3391
3392
3393
3394
3395
3396
3397
3398
3398
3399
3399
3400
3401
3402
3403
3404
3405
3406
3407
3408
3409
3409
3410
3411
3412
3413
3414
3415
3416
3417
3418
3419
3419
3420
3421
3422
3423
3424
3425
3426
3427
3428
3429
3429
3430
3431
3432
3433
3434
3435
3436
3437
3438
3439
3439
3440
3441
3442
3443
3444
3445
3446
3447
3448
3449
3449
3450
3451
3452
3453
3454
3455
3456
3457
3458
3459
3459
3460
3461
3462
3463
3464
3465
3466
3467
3468
3469
3469
3470
3471
3472
3473
3474
3475
3476
3477
3478
3479
3479
3480
3481
3482
3483
3484
3485
3486
3487
3488
3489
3489
3490
3491
3492
3493
3494
3495
3496
3497
3498
3498
3499
3499
3500
3501
3502
3503
3504
3505
3506
3507
3508
3509
3509
3510
3511
3512
3513
3514
3515
3516
3517
3518
3519
3519
3520
3521
3522
3523
3524
3525
3526
3527
3528
3529
3529
3530
3531
3532
3533
3534
3535
3536
3537
3538
3539
3539
3540
3541
3542
3543
3544
3545
3546
3547
3548
3549
3549
3550
3551
3552
3553
3554
3555
3556
3557
3558
3559
3559
3560
3561
3562
3563
3564
3565
3566
3567
3568
3569
3569
3570
3571
3572
3573
3574
3575
3576
3577
3578
3579
3579
3580
3581
3582
3583
3584
3585
3586
3587
3588
3589
3589
3590
3591
3592
3593
3594
3595
3596
3597
3598
3598
3599
3599
3600
3601
3602
3603
3604
3605
3606
3607
3608
3609
3609
3610
3611
3612
3613
3614
3615
3616
3617
3618
3619
3619
3620
3621
3622
3623
3624
3625
3626
3627
3628
3629
3629
3630
3631
3632
3633
3634
3635
3636
3637
3638
3639
3639
3640
3641
3642
3643
3644
3645
3646
3647
3648
3649
3649
3650
3651
3652
3653
3654
3655
3656
3657
3658
3659
3659
3660
3661
3662
3663
3664
3665
3666
3667
3668
3669
3669
3670
3671
3672
3673
3674
3675
3676
3677
3678
3679
3679
3680
3681
3682
3683
3684
3685
3686
3687
3688
3689
3689
3690
3691
3692
3693
3694
3695
3696
3697
3698
3698
3699
3699
3700
3701
3702
3703
3704
3705
3706
3707
3708
3709
3709
3710
3711
3712
3713
3714
3715
3716
3717
3718
3719
3719
3720
3721
3722
3723
3724
3725
3726
3727
3728
3729
3729
3730
3731
3732
3733
3734
3735
3736
3737
3738
3739
3739
3740
3741
3742
3743
3744
3745
3746
3747
3748
3749
3749
3750
3751
3752
3753
3754
3755
3756
3757
3758
3759
3759
3760
3761
3762
3763
3764
3765
3766
3767
3768
3769
3769
3770
3771
3772
3773
3774
3775
3776
3777
3778
3779
3779
3780
3781
3782
3783
3784
3785
3786
3787
3788
3789
3789
3790
3791
3792
3793
3794
3795
3796
3797
3798
3798
3799
```

The screenshot shows a Microsoft Visual Studio interface with the following details:

- Title Bar:** Debug Any CPU IIS Express (Google Chrome)
- Toolbars:** Standard, Debugging, Task List, Solution Explorer, Toolbox, Properties, Task List, Status Bar.
- Code Editor:** The main window displays the `MRAController.cs` file. The code is a C# class definition for `UQP.Controllers.MRAController`. The code handles data retrieval and calculation of provider percentages based on various filters like year, health plan ID, and provider ID.
- Code Snippet:**

```
1857     && x.ProjectectedScoreYear == inputDate.year)
1858     .Sum(x => x.ProjectectedMemberCount);

1859
1860
1861     List<MRAprevelance> prevelancelist = new List<MRAprevelance>();
1862     foreach (var per in providerpercent)
1863     {
1864         MRAprevelance mraobj = new MRAprevelance();
1865
1866         mraobj.providerid = per.provider_id;
1867         mraobj.hcc = per.value;
1868         mraobj.providerHCCmembers = per.membercount;
1869         mraobj.providerTotalmembers = overallprovidermember;
1870         double? overallhccmember = result.Where(x => x.value == per.value && x.HealthPlanID == inputDate.healthplanid)
1871             .Select(x => x.membercount).Sum();
1872
1873
1874         mraobj.overallHCCmembers = overallhccmember;
1875         mraobj.overallTotalmembers = overallmember;
1876
1877         mraobj.providerpercent = ((mraobj.providerHCCmembers * 100) / (overallprovidermember));
1878         if(double.IsInfinity(mraobj.providerpercent.Value))
1879         {
1880             mraobj.providerpercent = 0;
1881         }
1882
1883         mraobj.overallpercent = (overallhccmember * 100) / (overallmember);
1884         if (double.IsInfinity(mraobj.overallpercent.Value))
1885         {
1886             mraobj.overallpercent = 0;
1887         }
1888         mraobj.withoutHCCpercent = (100 - mraobj.providerpercent);
1889         if (double.IsInfinity(mraobj.withoutHCCpercent.Value))
1890         {
1891             mraobj.withoutHCCpercent = 0;
1892         }
1893     }
```
- Status Bar:** Shows "100 %"

UOP - Microsoft Visual Studio

File Edit View Project Build Debug Team Tools Test Analyze Window Help

Debug Any CPU IIS Express (Google Chrome)

MRAController.cs MembersController.cs Source Control Explorer ACO\_MembersMart.cs Notification\_UserSpecific.cs MRAfollowup(PagingParameterModel pagingParameterModel, InputData

UOP

```
1881 }  
1882  
1883 mraobj.overallpercent = (overallhccmember * 100) / (overallmember);  
1884 if (double.IsInfinity(mraobj.overallpercent.Value))  
1885 {  
1886     mraobj.overallpercent = 0;  
1887 }  
1888 mraobj.withoutHCCpercent = (100 - mraobj.providerpercent);  
1889 if (double.IsInfinity(mraobj.withoutHCCpercent.Value))  
1890 {  
1891     mraobj.withoutHCCpercent = 0;  
1892 }  
1893  
1894 prevelancelist.Add(mraobj);  
1895  
1896  
1897  
1898 }  
1899 if (inputDate.report == false)  
1900 {  
1901     #region pagination  
1902     // Get's No of Rows Count  
1903     int count = (prevelancelist.Count < 16) ? prevelancelist.Count : 15;  
1904     //int count = lstmembermart.Count();  
1905  
1906     // Parameter is passed from Query string if it is null then it default Value will be pageNumber:1  
1907     int CurrentPage = inputDate.pageNumber;  
1908  
1909     // Parameter is passed from Query string if it is null then it default Value will be pageSize:20  
1910     int PageSize = inputDate.pageSize;  
1911  
1912     // Display TotalCount to Records to User  
1913     int TotalCount = count;  
1914  
1915     // Calculating Totalpage by Dividing (No of Records / Pagesize)  
1916     // TotalPage = TotalCount / PageSize  
1917 }
```

Debug Any CPU IIS Express (Google Chrome) UQP

UQP.Controllers.MRAController.cs

```
1914 int totalCount = count;
1915
1916     // Calculating Totalpage by Dividing (No of Records / Pagesize)
1917     int totalPages = (int)Math.Ceiling(count / (double)pageSize);
1918
1919     // Returns List of Customer after applying Paging
1920     var items = prevelancelist.OrderByDescending(x => x.overallpercent).Take(15).ToList();
1921
1922
1923     // if CurrentPage is greater than 1 means it has previousPage
1924     var previousPage = currentPage > 1 ? "Yes" : "No";
1925
1926     // if TotalPages is greater than currentPage means it has nextPage
1927     var nextPage = currentPage < TotalPages ? "Yes" : "No";
1928
1929
1930     // Object which we are going to send in header
1931     var paginationMetadata = new
1932     {
1933         totalCount = totalCount,
1934         pageSize = pageSize,
1935         currentPage = currentPage,
1936         totalPages = totalPages,
1937         previousPage,
1938         nextPage
1939     };
1940
1941     // Setting Header
1942
1943     HttpContext.Current.Response.Headers.Add("Paging-Headers", Newtonsoft.Json.JsonConvert.SerializeObject(paginationMetadata));
1944     //HttpContext.Current.Response.Headers.Add("Filename", zipFileName);
1945     //HttpContext.Current.Response.Headers.Add("Access-Control-Expose-Headers", "Paging-Headers");
1946     #endregion
1947
1948     return Request.CreateResponse(HttpStatusCode.OK, items);
1949 }
```

Output

The screenshot shows a Microsoft Visual Studio interface with the following details:

- Open Files:** UOP.Controllers.MRAController.cs
- Toolbars:** Standard, Ruler, Status Bar
- Toolbox:** Standard
- Tool Windows:** Output
- Status Bar:** Scanned with CamScanner

```
1926 // if TotalPages is greater than CurrentPage means it has nextPage
1927 var nextPage = currentPage < TotalPages ? "Yes" : "No";
1928
1929 // Object which we are going to send in header
1930 var paginationMetadata = new
1931 {
1932     totalCount = TotalCount,
1933     pageSize = PageSize,
1934     currentPage = currentPage,
1935     totalPages = TotalPages,
1936     previousPage,
1937     nextPage
1938 };
1939
1940 // Setting Header
1941
1942 HttpContext.Current.Response.Headers.Add("Paging-Headers", Newtonsoft.Json.JsonConvert.SerializeObject(paginationMetadata));
1943 //HttpContext.Current.Response.Headers.Add("Filename", zipFileName);
1944 HttpContext.Current.Response.Headers.Add("Access-Control-Expose-Headers", "Paging-Headers");
1945 #endregion
1946
1947 return Request.CreateResponse(HttpStatusCode.OK, items);
1948 }
1949 else
1950 {
1951     var rpoj = new ReportsController
1952     {
1953         Request = new System.Net.Http.HttpRequestMessage(),
1954         Configuration = new HttpConfiguration()
1955     };
1956     return rpoj.GenerateNotifReport(null, null, null, null, null, null, "MRA SCORECARD", "excel", null, null, null, null, null,
1957         null, null, null, null, null, null, null, null, prevelanceList.OrderByDescending(x => x.overallpercent).Take(15).ToList(), null);
1958 }
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2498
2499
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2598
2599
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2619
2620
2621
2622
2623
2624
2625
2626
2627
2628
2629
2629
2630
2631
2632
2633
2634
2635
2636
2637
2638
2639
2639
2640
2641
2642
2643
2644
2645
2646
2647
2648
2649
2649
2650
2651
2652
2653
2654
2655
2656
2657
2658
2659
2659
2660
2661
2662
2663
2664
2665
2666
2667
2668
2669
2669
2670
2671
2672
2673
2674
2675
2676
2677
2678
2679
2679
2680
2681
2682
2683
2684
2685
2686
2687
2688
2689
2689
2690
2691
2692
2693
2694
2695
2696
2697
2698
2698
2699
2699
2700
2701
2702
2703
2704
2705
2706
2707
2708
2709
2709
2710
2711
2712
2713
2714
2715
2716
2717
2718
2719
2719
2720
2721
2722
2723
2724
2725
2726
2727
2728
2729
2729
2730
2731
2732
2733
2734
2735
2736
2737
2738
2739
2739
2740
2741
2742
2743
2744
2745
2746
2747
2748
2749
2749
2750
2751
2752
2753
2754
2755
2756
2757
2758
2759
2759
2760
2761
2762
2763
2764
2765
2766
2767
2768
2769
2769
2770
2771
2772
2773
2774
2775
2776
2777
2778
2779
2779
2780
2781
2782
2783
2784
2785
2786
2787
2788
2789
2789
2790
2791
2792
2793
2794
2795
2796
2797
2798
2798
2799
2799
2800
2801
2802
2803
2804
2805
2806
2807
2808
2809
2809
2810
2811
2812
2813
2814
2815
2816
2817
2818
2819
2819
2820
2821
2822
2823
2824
2825
2826
2827
2828
2829
2829
2830
2831
2832
2833
2834
2835
2836
2837
2838
2839
2839
2840
2841
2842
2843
2844
2845
2846
2847
2848
2849
2849
2850
2851
2852
2853
2854
2855
2856
2857
2858
2859
2859
2860
2861
2862
2863
2864
2865
2866
2867
2868
2869
2869
2870
2871
2872
2873
2874
2875
2876
2877
2878
2879
2879
2880
2881
2882
2883
2884
2885
2886
2887
2888
2889
2889
2890
2891
2892
2893
2894
2895
2896
2897
2898
2898
2899
2899
2900
2901
2902
2903
2904
2905
2906
2907
2908
2909
2909
2910
2911
2912
2913
2914
2915
2916
2917
2918
2919
2919
2920
2921
2922
2923
2924
2925
2926
2927
2928
2929
2929
2930
2931
2932
2933
2934
2935
2936
2937
2938
2939
2939
2940
2941
2942
2943
2944
2945
2946
2947
2948
2949
2949
2950
2951
2952
2953
2954
2955
2956
2957
2958
2959
2959
2960
2961
2962
2963
2964
2965
2966
2967
2968
2969
2969
2970
2971
2972
2973
2974
2975
2976
2977
2978
2979
2979
2980
2981
2982
2983
2984
2985
2986
2987
2988
2989
2989
2990
2991
2992
2993
2994
2995
2996
2997
2998
2998
2999
2999
3000
3001
3002
3003
3004
3005
3006
3007
3008
3009
3009
3010
3011
3012
3013
3014
3015
3016
3017
3018
3019
3019
3020
3021
3022
3023
3024
3025
3026
3027
3028
3029
3029
3030
3031
3032
3033
3034
3035
3036
3037
3038
3038
3039
3039
3040
3041
3042
3043
3044
3045
3046
3047
3048
3049
3049
3050
3051
3052
3053
3054
3055
3056
3057
3058
3059
3059
3060
3061
3062
3063
3064
3065
3066
3067
3068
3069
3069
3070
3071
3072
3073
3074
3075
3076
3077
3078
3079
3079
3080
3081
3082
3083
3084
3085
3086
3087
3088
3089
3089
3090
3091
3092
3093
3094
3095
3096
3097
3098
3098
3099
3099
3100
3101
3102
3103
3104
3105
3106
3107
3108
3109
3109
3110
3111
3112
3113
3114
3115
3116
3117
3118
3119
3119
3120
3121
3122
3123
3124
3125
3126
3127
3128
3129
3129
3130
3131
3132
3133
3134
3135
3136
3137
3138
3139
3139
3140
3141
3142
3143
3144
3145
3146
3147
3148
3149
3149
3150
3151
3152
3153
3154
3155
3156
3157
3158
3159
3159
3160
3161
3162
3163
3164
3165
3166
3167
3168
3169
3169
3170
3171
3172
3173
3174
3175
3176
3177
3178
3179
3179
3180
3181
3182
3183
3184
3185
3186
3187
3188
3189
3189
3190
3191
3192
3193
3194
3195
3196
3197
3197
3198
3199
3199
3200
3201
3202
3203
3204
3205
3206
3207
3208
3209
3209
3210
3211
3212
3213
3214
3215
3216
3217
3218
3219
3219
3220
3221
3222
3223
3224
3225
3226
3227
3228
3229
3229
3230
3231
3232
3233
3234
3235
3236
3237
3238
3239
3239
3240
3241
3242
3243
3244
3245
3246
3247
3248
3249
3249
3250
3251
3252
3253
3254
3255
3256
3257
3258
3259
3259
3260
3261
3262
3263
3264
3265
3266
3267
3268
3269
3269
3270
3271
3272
3273
3274
3275
3276
3277
3278
3279
3279
3280
3281
3282
3283
3284
3285
3286
3287
3288
3289
3289
3290
3291
3292
3293
3294
3295
3296
3297
3297
3298
3299
3299
3300
3301
3302
3303
3304
3305
3306
3307
3308
3309
3309
3310
3311
3312
3313
3314
3315
3316
3317
3318
3319
3319
3320
3321
3322
3323
3324
3325
3326
3327
3328
3329
3329
3330
3331
3332
3333
3334
3335
3336
3337
3338
3339
3339
3340
3341
3342
3343
3344
3345
3346
3347
3348
3349
3349
3350
3351
3352
3353
3354
3355
3356
3357
3358
3359
3359
3360
3361
3362
3363
3364
3365
3366
3367
3368
3369
3369
3370
3371
3372
3373
3374
3375
3376
3377
3378
3379
3379
3380
3381
3382
3383
3384
3385
3386
3387
3388
3389
3389
3390
3391
3392
3393
3394
3395
3396
3397
3397
3398
3399
3399
3400
3401
3402
3403
3404
3405
3406
3407
3408
3409
3409
3410
3411
3412
3413
3414
3415
3416
3417
3418
3419
3419
3420
3421
3422
3423
3424
3425
3426
3427
3428
3429
3429
3430
3431
3432
3433
3434
3435
3436
3437
3438
3439
3439
3440
3441
3442
3443
3444
3445
3446
3447
3448
3449
3449
3450
3451
3452
3453
3454
3455
3456
3457
3458
3459
3459
3460
3461
3462
3463
3464
3465
3466
3467
3468
3469
3469
3470
3471
3472
3473
3474
3475
3476
3477
3478
3479
3479
3480
3481
3482
3483
3484
3485
3486
3487
3488
3489
3489
3490
3491
3492
3493
3494
3495
3496
3497
3497
3498
3499
3499
3500
3501
3502
3503
3504
3505
3506
3507
3508
3509
3509
3510
3511
3512
3513
3514
3515
3516
3517
3518
3519
3519
3520
3521
3522
3523
3524
3525
3526
3527
3528
3529
3529
3530
3531
3532
3533
3534
3535
3536
3537
3538
3539
3539
3540
3541
3542
3543
3544
3545
3546
3547
3548
3549
3549
3550
3551
3552
3553
3554
3555
3556
3557
3558
3559
3559
3560
3561
3562
3563
3564
3565
3566
3567
3568
3569
3569
3570
3571
3572
3573
3574
3575
3576
3577
3578
3579
3579
3580
3581
3582
3583
3584
3585
3586
3587
3588
3589
3589
3590
3591
3592
3593
3594
3595
3596
3597
3598
3598
3599
3599
3600
3601
3602
3603
3604
3605
3606
3607
3608
3609
3609
3610
3611
3612
3613
3614
3615
3616
3617
3618
3619
3619
3620
3621
3622
3623
3624
3625
3626
3627
3628
3629
3629
3630
3631
3632
3633
3634
3635
3636
3637
3638
3639
3639
3640
3641
3642
3643
3644
3645
3646
3647
3648
3649
3649
3650
3651
3652
3653
3654
3655
3656
3657
3658
3659
3659
3660
3661
3662
3663
3664
3665
3666
3667
3668
3669
3669
3670
3671
3672
3673
3674
3675
3676
3677
3678
3679
3679
3680
3681
3682
3683
3684
3685
3686
3687
3688
3689
3689
3690
3691
3692
3693
3694
3695
3696
3697
3698
3698
3699
3699
3700
3701
3702
3703
3704
3705
3706
3707
3708
3709
3709
3710
3711
3712
3713
3714
3715
3716
3717
3718
3719
3719
3720
3721
3722
3723
3724
3725
3726
3727
3728
3729
3729
3730
3731
3732
3733
3734
3735
3736
3737
3738
3739
3739
3740
3741
3742
3743
3744
3745
3746
3747
3748
3749
3749
3750
3751
3752
3753
3754
3755
3756
3757
3758
3759
3759
3760
3761
3762
3763
3764
3765
3766
3767
3768
3769
3769
3770
3771
3772
3773
3774
3775
3776
3777
3778
3779
3779
3780
3781
3782
3783
3784
3785
3786
3787
3788
3789
3789
3790
3791
3792
3793
3794
3795
3796
3797
3797
3798
3799
3799
3800
3801
3802
3803
3804
3805
3806
3807
3808
3809
3809
3810
3811
3812
3813
3814
3815
3816
3817
3818
3819
3819
3820
3821
3822
3823
3824
3825
3826
3827
3828
3829
3829
3830
3831
3832
3833
3834
3835
3836
3837
3838
3839
3839
3840
3841
3842
3843
3844
3845
3846
3847
3848
3849
3849
3850
3851
3852
3853
3854
3855
3856
3857
3858
3859
3859
3860
3861
3862
3863
3864
3865
3866
3867
3868
3869
3869
3870
3871
3872
3873
3874
3875
3876
3877
3878
3879
3879
3880
3881
3882
3883
3884
3885
3886
3887
3888
3889
3889
3890
3891
3892
3893
3894
3895
3896
3897

```

A screenshot of a Microsoft Visual Studio IDE interface. The main window shows a code editor with C# code for a controller. The code includes several catch blocks for exceptions. The status bar at the bottom indicates "Output" and "Show output from: Source Control - Team Foundation".

```
1959     }
1960     }
1961     }
1962     }
1963     }
1964     }
1965     }
1966     }
1967     }
1968     }
1969     }
1970     }
1971     catch (Exception e)
1972     {
1973         return Request.CreateResponse(e.InnerException);
1974     }
1975     }
1976     }
1977     }
1978     }
1979     }
1980     }
1981 }
```