**2.3. ПРОХОЖДЕНИЕ ДОРОГИ (УЧАСТКА) ПО АДМИНИСТРАТИВНО-ТЕРРИТОРИАЛЬНЫМ ОБРАЗОВАНИЯМ**

| **Наименование региона** | **Наименование района** | **Начало границы, км+** | **Конец границы, км+** | **Протяженность, км** |
| --- | --- | --- | --- | --- |
| **[CONTENT]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*/3805/\*$\*/  SELECT @NumRoad = NumRoad FROM ListDataSources WHERE id\_ = @NumDataSource  SELECT 'Иркутская область' [Наименование региона], rtrim(D.FullTitle) [Наименование района],  dbo.pp\_Km\_mFormat(LRP.StartPos) [Начало границы, км+], dbo.pp\_Km\_mFormat(LRP.EndPos) [Конец границы, км+],  convert(numeric(10,3), round((SELECT sum(t.EndPos - t.StartPos) FROM ListRoadParts t WHERE t.NumRoad = LRP.NumRoad and t.NumDataSource = LRP.NumDataSource and t.NumPartType = 13)/1000.0, 3)) [Протяженность, км]  FROM ListRoadParts LRP  JOIN ListRoads R ON R.id\_ = LRP.NumRoad  JOIN ListDistricts D ON D.id\_ = R.NumDistrict  WHERE LRP.NumRoad = @NumRoad  and LRP.NumDataSource = @NumDataSource  and LRP.NumPartType = 33 | | | | |

**2.4. ПРОТЯЖЕННОСТЬ ДОРОГИ (УЧАСТКА)**

| **Начало дороги (участка), км** | **Конец дороги (участка), км** | **Общая протяженность, км** | | | **В том числе участков** | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **дороги**  **(участка)** | **подъездов**  **(объездов)** | **дороги вместе с**  **подъездами** | **обслуживаемых**  **дорожной организацией** | **находящихся в**  **ведении городов** | **совмещенных** |
| **[CONTENT]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*/3805/\*$\*/  SELECT @NumRoad = NumRoad FROM ListDataSources WHERE id\_ = @NumDataSource  SELECT dbo.pp\_Km\_mFormat(LRP.StartPos) [Начало дороги (участка), км], dbo.pp\_Km\_mFormat(LRP.EndPos) [Конец дороги (участка), км],  convert(numeric(10,3), round((SELECT sum(t.EndPos - t.StartPos) FROM ListRoadParts t WHERE t.NumRoad = LRP.NumRoad and t.NumDataSource = LRP.NumDataSource and t.NumPartType = 13)/1000.0, 3)) [Общая протяженность, км],  0 [подъездов (объездов)],  convert(numeric(10,3), round((SELECT sum(t.EndPos - t.StartPos) FROM ListRoadParts t WHERE t.NumRoad = LRP.NumRoad and t.NumDataSource = LRP.NumDataSource and t.NumPartType = 13)/1000.0, 3)) [дороги вместе с подъездами],  convert(numeric(10,3), round((SELECT sum(t.EndPos - t.StartPos) FROM ListRoadParts t WHERE t.NumRoad = LRP.NumRoad and t.NumDataSource = LRP.NumDataSource and t.NumPartType = 13)/1000.0, 3)) [обслуживаемых дорожной организацией],  0 [находящихся в ведении городов],  0 [совмещенных]  FROM ListRoadParts LRP  JOIN ListRoads R ON R.id\_ = LRP.NumRoad  JOIN ListDistricts D ON D.id\_ = R.NumDistrict  WHERE LRP.NumRoad = @NumRoad  and LRP.NumDataSource = @NumDataSource  and LRP.NumPartType = 33 | | | | | | | |

**2.9. ТАБЛИЦА ОСНОВНЫХ РАССТОЯНИЙ (в целых километрах)**

| **[CONTENT][AUTO\_EXPAND]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*/3844/\*$\*/  SELECT @NumRoad = NumRoad FROM ListDataSources WHERE id\_ = @NumDataSource  DECLARE @NP table (num int identity(1,1), StartPos int, EndPos int, FullTitle varchar(1000), Displacement int)  INSERT INTO @NP (StartPos, EndPos, FullTitle, Displacement)  SELECT LRP.StartPos, LRP.EndPos, rtrim(I.FullTitle) FullTitle,  convert(int, round(LRP.StartPos + (LRP.EndPos - LRP.StartPos)/2, 0)) Displacement  FROM ListRoadParts LRP  JOIN ListInhabitedLocalities I ON I.id\_ = LRP.Reference  WHERE LRP.NumRoad = @NumRoad  and LRP.NumDataSource = @NumDataSource  and LRP.NumPartType = 9  ORDER BY Displacement  SELECT t.\*,  t1.num num1,  t1.FullTitle FullTitle1,  (case when t1.Displacement = t.Displacement  then '\*\*\*'  else convert(varchar(10), convert(int,  round((dbo.pp\_GetMaxValue(t.Displacement, t1.Displacement) - dbo.pp\_GetMinValue(t.Displacement, t1.Displacement))/1000.0, 0)  ))  end) km  INTO #npkm  FROM @np t  CROSS JOIN @np t1  SELECT km, l0, npp, FullTitle,  km1, km2, km3, km4, km5, km6, km7, km8, km9, km10  FROM (  SELECT 0 OrderFld,  'Километраж от начала дороги' km,  'Протяж. уч-ков, находящихся в ведении н.п.' l0,  '№' npp,  'Наименование н.п.' FullTitle,  max(case when t.num1 = 1 then t.FullTitle1 else '' end) COLLATE DATABASE\_DEFAULT as km1,  max(case when t.num1 = 2 then t.FullTitle1 else '' end) COLLATE DATABASE\_DEFAULT as km2,  max(case when t.num1 = 3 then t.FullTitle1 else '' end) COLLATE DATABASE\_DEFAULT as km3,  max(case when t.num1 = 4 then t.FullTitle1 else '' end) COLLATE DATABASE\_DEFAULT as km4,  max(case when t.num1 = 5 then t.FullTitle1 else '' end) COLLATE DATABASE\_DEFAULT as km5,  max(case when t.num1 = 6 then t.FullTitle1 else '' end) COLLATE DATABASE\_DEFAULT as km6,  max(case when t.num1 = 7 then t.FullTitle1 else '' end) COLLATE DATABASE\_DEFAULT as km7,  max(case when t.num1 = 8 then t.FullTitle1 else '' end) COLLATE DATABASE\_DEFAULT as km8,  max(case when t.num1 = 9 then t.FullTitle1 else '' end) COLLATE DATABASE\_DEFAULT as km9,  max(case when t.num1 = 10 then t.FullTitle1 else '' end) COLLATE DATABASE\_DEFAULT as km10  FROM #npkm t  UNION ALL  SELECT t.num,  cast(t.Displacement as varchar),  '0',  cast(t.num as varchar) npp,  t.FullTitle,  max(case when t.num1 = 1 then t.km else '' end) COLLATE DATABASE\_DEFAULT,  max(case when t.num1 = 2 then t.km else '' end) COLLATE DATABASE\_DEFAULT,  max(case when t.num1 = 3 then t.km else '' end) COLLATE DATABASE\_DEFAULT,  max(case when t.num1 = 4 then t.km else '' end) COLLATE DATABASE\_DEFAULT,  max(case when t.num1 = 5 then t.km else '' end) COLLATE DATABASE\_DEFAULT,  max(case when t.num1 = 6 then t.km else '' end) COLLATE DATABASE\_DEFAULT,  max(case when t.num1 = 7 then t.km else '' end) COLLATE DATABASE\_DEFAULT,  max(case when t.num1 = 8 then t.km else '' end) COLLATE DATABASE\_DEFAULT,  max(case when t.num1 = 9 then t.km else '' end) COLLATE DATABASE\_DEFAULT,  max(case when t.num1 = 10 then t.km else '' end) COLLATE DATABASE\_DEFAULT  FROM #npkm t  GROUP BY t.num, t.FullTitle, t.Displacement  ) T  ORDER BY OrderFld  DROP TABLE #npkm |
| --- |

**4.2. ШИРИНА ЗЕМЛЯНОГО ПОЛОТНА**

| **На 1 января** | **Протяженность (км) при ширине (м)** | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| **меньше 8** | **от 8.0 до 9.9** | **от 10.0 до 11.9** | **от 12.0 до 14.9** | **от 15.0 до 27.4** | **27.5 и более** |
| **[CONTENT]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*/3805/\*$\*/  SELECT @NumRoad = NumRoad FROM ListDataSources WHERE id\_ = @NumDataSource  DECLARE @year int  SET @year = 2015  SELECT @year + 1 YR INTO #YRS UNION ALL  SELECT @year + 2 YR UNION ALL  SELECT @year + 3 YR UNION ALL  SELECT @year + 4 YR UNION ALL  SELECT @year + 5 YR UNION ALL  SELECT @year + 6 YR UNION ALL  SELECT @year + 7 YR UNION ALL  SELECT @year + 8 YR UNION ALL  SELECT @year + 9 YR  SELECT round(StartPos, 0) StartPos, round(EndPos, 0) EndPos, Value2 Width  INTO #Road  FROM ListRoadParts  WHERE NumRoad = @NumRoad  and NumDataSource = @NumDataSource  and NumPartType = 14  SELECT round(StartPos, 0) StartPos, round(EndPos, 0) EndPos, NumPlace, Width  INTO #Sides  FROM ListRoadSides  WHERE NumRoad = @NumRoad  and NumDataSource = @NumDataSource  DECLARE @Positions table (id\_ int identity(1,1), Pos int)  INSERT INTO @Positions (Pos)  SELECT StartPos  FROM #Sides  UNION  SELECT EndPos  FROM #Sides  UNION  SELECT StartPos  FROM #Road  UNION  SELECT EndPos  FROM #Road  ORDER BY 1  SELECT P.Pos StartPos, P1.Pos EndPos  INTO #Parts  FROM @Positions P  JOIN @Positions P1 ON P.id\_ + 1 = P1.id\_  SELECT P.StartPos, P.EndPos, LRP.Width + S257.Width + S258.Width Width  INTO #RoadFullWidthes  FROM #Parts P  LEFT JOIN #Road LRP ON (P.StartPos >= LRP.StartPos and P.StartPos < LRP.EndPos)  LEFT JOIN #Sides S257 ON (P.StartPos >= S257.StartPos and P.StartPos < S257.EndPos)  and S257.NumPlace = 257  LEFT JOIN #Sides S258 ON (P.StartPos >= S258.StartPos and P.StartPos < S258.EndPos)  and S258.NumPlace = 258  SELECT @year YR,  ISNULL(convert(varchar(10), NULLIF(convert(numeric(10,3),round(  sum(case when LRP.Width < 8 then LRP.EndPos - LRP.StartPos else 0 end)/1000.0  , 3)), 0)), ''),  ISNULL(convert(varchar(10), NULLIF(convert(numeric(10,3),round(  sum(case when LRP.Width >= 8 and LRP.Width < 10 then LRP.EndPos - LRP.StartPos else 0 end)/1000.0  , 3)), 0)), ''),  ISNULL(convert(varchar(10), NULLIF(convert(numeric(10,3),round(  sum(case when LRP.Width >= 10 and LRP.Width < 12 then LRP.EndPos - LRP.StartPos else 0 end)/1000.0  , 3)), 0)), ''),  ISNULL(convert(varchar(10), NULLIF(convert(numeric(10,3),round(  sum(case when LRP.Width >= 12 and LRP.Width < 15 then LRP.EndPos - LRP.StartPos else 0 end)/1000.0  , 3)), 0)), ''),  ISNULL(convert(varchar(10), NULLIF(convert(numeric(10,3),round(  sum(case when LRP.Width >= 15 and LRP.Width < 27.5 then LRP.EndPos - LRP.StartPos else 0 end)/1000.0  , 3)), 0)), ''),  ISNULL(convert(varchar(10), NULLIF(convert(numeric(10,3),round(  sum(case when LRP.Width >= 27.5 then LRP.EndPos - LRP.StartPos else 0 end)/1000.0  , 3)), 0)), '')  FROM #RoadFullWidthes LRP  UNION ALL  SELECT YR,  '', '', '', '', '', ''  FROM #YRS  ORDER BY YR  DROP TABLE #YRS  DROP TABLE #Road  DROP TABLE #Sides  DROP TABLE #Parts  DROP TABLE #RoadFullWidthes | | | | | | |

**4.3.1. ШИРИНА ПРОЕЗЖЕЙ ЧАСТИ**

| **На 1-ое января** | **Протяженность (км) при ширине (м)** | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **до 4** | **от 4.1 до 4.4** | **от 4.5 до 5.9** | **от 6.0 до 6.5** | **от 6.6 до 6.9** | **от 7.0 до 7.4** | **от 7.5 до 9.9** | **10.0 – 15.0** | **15.1 – 27.0** | **более 27.0** |
| **[CONTENT]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*/3805/\*$\*/  SELECT @NumRoad = NumRoad FROM ListDataSources WHERE id\_ = @NumDataSource  DECLARE @year int  SET @year = 2015  SELECT @year + 1 YR INTO #YRS UNION ALL  SELECT @year + 2 YR UNION ALL  SELECT @year + 3 YR UNION ALL  SELECT @year + 4 YR UNION ALL  SELECT @year + 5 YR UNION ALL  SELECT @year + 6 YR UNION ALL  SELECT @year + 7 YR UNION ALL  SELECT @year + 8 YR UNION ALL  SELECT @year + 9 YR  SELECT @year YR,  ISNULL(convert(varchar(10), NULLIF(convert(numeric(10,3),round(  sum(case when LRP.Value2 < 4.1 then LRP.EndPos - LRP.StartPos else 0 end)/1000.0  , 3)), 0)), ''),  ISNULL(convert(varchar(10), NULLIF(convert(numeric(10,3),round(  sum(case when LRP.Value2 >= 4.1 and LRP.Value2 < 4.5 then LRP.EndPos - LRP.StartPos else 0 end)/1000.0  , 3)), 0)), ''),  ISNULL(convert(varchar(10), NULLIF(convert(numeric(10,3),round(  sum(case when LRP.Value2 >= 4.5 and LRP.Value2 < 6 then LRP.EndPos - LRP.StartPos else 0 end)/1000.0  , 3)), 0)), ''),  ISNULL(convert(varchar(10), NULLIF(convert(numeric(10,3),round(  sum(case when LRP.Value2 >= 6 and LRP.Value2 < 6.6 then LRP.EndPos - LRP.StartPos else 0 end)/1000.0  , 3)), 0)), ''),  ISNULL(convert(varchar(10), NULLIF(convert(numeric(10,3),round(  sum(case when LRP.Value2 >= 6.6 and LRP.Value2 < 7 then LRP.EndPos - LRP.StartPos else 0 end)/1000.0  , 3)), 0)), ''),  ISNULL(convert(varchar(10), NULLIF(convert(numeric(10,3),round(  sum(case when LRP.Value2 >= 7 and LRP.Value2 < 7.5 then LRP.EndPos - LRP.StartPos else 0 end)/1000.0  , 3)), 0)), ''),  ISNULL(convert(varchar(10), NULLIF(convert(numeric(10,3),round(  sum(case when LRP.Value2 >= 7.5 and LRP.Value2 < 10 then LRP.EndPos - LRP.StartPos else 0 end)/1000.0  , 3)), 0)), ''),  ISNULL(convert(varchar(10), NULLIF(convert(numeric(10,3),round(  sum(case when LRP.Value2 >= 10 and LRP.Value2 < 15.1 then LRP.EndPos - LRP.StartPos else 0 end)/1000.0  , 3)), 0)), ''),  ISNULL(convert(varchar(10), NULLIF(convert(numeric(10,3),round(  sum(case when LRP.Value2 >= 15.1 and LRP.Value2 < 27 then LRP.EndPos - LRP.StartPos else 0 end)/1000.0  , 3)), 0)), ''),  ISNULL(convert(varchar(10), NULLIF(convert(numeric(10,3),round(  sum(case when LRP.Value2 >= 27 then LRP.EndPos - LRP.StartPos else 0 end)/1000.0  , 3)), 0)), '')  FROM ListRoadParts LRP  WHERE LRP.NumRoad = @NumRoad  and LRP.NumDataSource = @NumDataSource  and LRP.NumPartType = 14  UNION ALL  SELECT YR,  '', '', '', '', '', '', '', '', '', ''  FROM #YRS  ORDER BY YR  DROP TABLE #YRS | | | | | | | | | | |

**4.3.2. ПРОТЯЖЕННОСТЬ И ПЛОЩАДЬ ПОКРЫТИЙ**

| **Типы покрытий** | **Протяженность (км) / площадь (м2) на 1 января** | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **2015г.** | **2016г.** | **2017г.** | **2018г.** | **2019г.** | **2020г.** | **2021г.** | **2022г.** | **2023г.** | **2024г.** |
| **[CONTENT]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*/3805/\*$\*/  SELECT @NumRoad = NumRoad FROM ListDataSources WHERE id\_ = @NumDataSource  SELECT round(StartPos, 0) StartPos, round(EndPos, 0) EndPos, Value2 Width  INTO #Road  FROM ListRoadParts  WHERE NumRoad = @NumRoad  and NumDataSource = @NumDataSource  and NumPartType = 14  SELECT round(StartPos, 0) StartPos, round(EndPos, 0) EndPos, Reference  INTO #Surf  FROM ListRoadParts  WHERE NumRoad = @NumRoad  and NumDataSource = @NumDataSource  and NumPartType = 13  DECLARE @Positions table (id\_ int identity(1,1), Pos int)  INSERT INTO @Positions (Pos)  SELECT StartPos  FROM #Surf  UNION  SELECT EndPos  FROM #Surf  UNION  SELECT StartPos  FROM #Road  UNION  SELECT EndPos  FROM #Road  ORDER BY 1  SELECT P.Pos StartPos, P1.Pos EndPos  INTO #Parts  FROM @Positions P  JOIN @Positions P1 ON P.id\_ + 1 = P1.id\_  SELECT SF.Reference,  convert(numeric(10,3), round(sum(P.EndPos - P.StartPos)/1000.0, 3)) Length,  convert(int, round(sum(LRP.Width \* (P.EndPos - P.StartPos)), 0)) Area  INTO #RoadAreas  FROM #Parts P  LEFT JOIN #Road LRP ON (P.StartPos >= LRP.StartPos and P.StartPos < LRP.EndPos)  JOIN #Surf SF ON (P.StartPos >= SF.StartPos and P.StartPos < SF.EndPos)  GROUP BY SF.Reference  SELECT '1. Усовeршенствованные', '', '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT ' Цементобетонные',  ISNULL((SELECT convert(varchar(10), Length) + '/' + convert(varchar(10), Area)  FROM #RoadAreas  WHERE Reference = 233), ''),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT ' Асфальтобетонные',  ISNULL((SELECT convert(varchar(10), Length) + '/' + convert(varchar(10), Area)  FROM #RoadAreas  WHERE Reference = 234), ''),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT ' Щебеночные (гравийные), обработанные вяжущими',  ISNULL((SELECT convert(varchar(10), Length) + '/' + convert(varchar(10), Area)  FROM #RoadAreas  WHERE Reference = 235), ''),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT ' Мостовые из мозаики и брусчатки',  '',  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT ' Прочие',  '',  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT '2.Переходные', '', '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT ' Щебеночные (гравийные)',  ISNULL((SELECT convert(varchar(10), Length) + '/' + convert(varchar(10), Area)  FROM #RoadAreas  WHERE Reference = 236), ''),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT ' Покрытия из грунтов и местных материалов, обработанных вяжущими',  ISNULL((SELECT convert(varchar(10), Length) + '/' + convert(varchar(10), Area)  FROM #RoadAreas  WHERE Reference = 237), ''),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT ' Прочие',  '',  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT '3.Низшие', '', '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT ' Грунтовые естественные',  ISNULL((SELECT convert(varchar(10), Length) + '/' + convert(varchar(10), Area)  FROM #RoadAreas  WHERE Reference = 238), ''),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT ' Прочие',  ISNULL((SELECT convert(varchar(10), Length) + '/' + convert(varchar(10), Area)  FROM #RoadAreas  WHERE ISNULL(Reference, 0) not in(233, 234,235,236,237,238)), ''),  '', '', '', '', '', '', '', '', '', ''  DROP TABLE #Surf  DROP TABLE #Road  DROP TABLE #Parts  DROP TABLE #RoadAreas | | | | | | | | | | |

**4.3.3. ПРОТЯЖЕННОСТЬ ПОКРЫТИЙ НА МОСТАХ (ПУТЕПРОВОДАХ)**

| **Типы покрытий** | **Протяженность (км) на 1 января** | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **2015г.** | **2016г.** | **2017г.** | **2018г.** | **2019г.** | **2020г.** | **2021г.** | **2022г.** | **2023г.** | **2024г.** |
| [CONTENT]  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*/3805/\*$\*/  SELECT @NumRoad = NumRoad FROM ListDataSources WHERE id\_ = @NumDataSource  SELECT LRP.Reference, convert(int, round(sum(Extent), 0)) Length  INTO #BridgesLength  FROM ListBridges B  LEFT JOIN ListRoadParts LRP ON LRP.NumRoad = B.NumRoad  and LRP.NumDataSource = B.NumDataSource  and LRP.NumPartType = 13  and B.Displacement between LRP.StartPos and LRP.EndPos  WHERE B.NumRoad = @NumRoad  and B.NumDataSource = @NumDataSource  GROUP BY LRP.Reference  SELECT 'Цементобетонные',  ISNULL((select convert(varchar(10), Length) from #BridgesLength where Reference = 233), ''),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT 'Асфальтобетонные',  ISNULL((select convert(varchar(10), Length) from #BridgesLength where Reference = 234), ''),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT 'Щебёночные (гравийные)',  ISNULL((select convert(varchar(10), Length) from #BridgesLength where Reference = 236), ''),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT 'Прочие',  ISNULL((select convert(varchar(10), sum(Length)) from #BridgesLength where Reference = 238 or ISNULL(Reference, 0) not in (233, 234, 236)), ''),  '', '', '', '', '', '', '', '', '', ''  DROP TABLE #BridgesLength | | | | | | | | | | |

**4.4. ХАРАКТЕРИСТИКА ПРОФИЛЯ И ТРАССЫ : РАДИУСЫ КРИВЫХ И УКЛОНЫ, НЕ ОТВЕЧАЮЩИЕ ТРЕБОВАНИЯМ СНиПов ДЛЯ ДАННОЙ КАТЕГОРИИ ДОРОГИ**

| **На 1 января** | **Количество кривых с радиусом менее допустимых, шт** | **Суммарная длина кривых с радиусами менее допустимых, км** | **Количество участков с уклонами, превышающим допустимые, шт** | **Длина участков с уклонами, превышающими допустимые, км** |
| --- | --- | --- | --- | --- |
| [CONTENT]  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*/3805/\*$\*/  SELECT @NumRoad = NumRoad FROM ListDataSources WHERE id\_ = @NumDataSource  DECLARE @year int  SET @year = 2015  SELECT @year + 1 YR INTO #YRS UNION ALL  SELECT @year + 2 YR UNION ALL  SELECT @year + 3 YR UNION ALL  SELECT @year + 4 YR UNION ALL  SELECT @year + 5 YR UNION ALL  SELECT @year + 6 YR UNION ALL  SELECT @year + 7 YR UNION ALL  SELECT @year + 8 YR UNION ALL  SELECT @year + 9 YR UNION ALL  SELECT @year + 9 YR  SELECT  sum(case when (LRP.reference2 = 2385042 or LRP.value4 = 2385042) and LRP.EndPos - LRP.StartPos > 15 then 1 else 0 end) num,  convert(int, round(sum(case when (LRP.reference2 = 2385042 or LRP.value4 = 2385042) and LRP.EndPos - LRP.StartPos > 15 then LRP.EndPos - LRP.StartPos else 0 end), 0)) len  INTO #Curvies  FROM ListRoadParts LRP  WHERE LRP.NumPartType = 34  and LRP.NumRoad = @NumRoad  and LRP.NumDataSource = @NumDataSource    SELECT  sum(case when LRP.reference = 2385042 then 1 else 0 end) num,  convert(int, round(sum(case when LRP.reference = 2385042 then LRP.EndPos - LRP.StartPos else 0 end), 0)) len  INTO #Slopes  FROM ListRoadParts LRP  WHERE LRP.NumPartType = 32  and LRP.NumRoad = @NumRoad  and LRP.NumDataSource = @NumDataSource  SELECT @year YR,  convert(varchar(10), C.num), convert(varchar(10), C.len),  convert(varchar(10), S.num), convert(varchar(10), S.len)  FROM #Curvies C  CROSS JOIN #Slopes S  UNION  SELECT YR,  '', '', '', ''  FROM #YRS  ORDER BY YR  DROP TABLE #YRS  DROP TABLE #Curvies  DROP TABLE #Slopes | | | | |

**4.6. ДОРОЖНЫЕ ИНЖЕНЕРНЫЕ УСТРОЙСТВА И ОБСТАНОВКА ДОРОГ**

| **Наименование** | **Наличие на 1 января** | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **2015г.** | **2016г.** | **2017г.** | **2018г.** | **2019г.** | **2020г.** | **2021г.** | **2022г.** | **2023г.** | **2024г.** |
| [CONTENT]  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*/3805/\*$\*/  SELECT @NumRoad = NumRoad FROM ListDataSources WHERE id\_ = @NumDataSource  SELECT \*  INTO #Lamps  FROM ListRoadLamps  WHERE NumRoad = @NumRoad  and NumDataSource = @NumDataSource  DECLARE @LeftBorderLamps table (id\_ int identity(1,1), Displacement int)  DECLARE @RightBorderLamps table (id\_ int identity(1,1), Displacement int)  INSERT INTO @LeftBorderLamps (Displacement)  SELECT L.Displacement  FROM #Lamps L  LEFT JOIN #Lamps L1 ON L.Displacement - L1.Displacement > 0 and L.Displacement - L1.Displacement < 50  WHERE L1.id\_ is NULL  ORDER BY 1  INSERT INTO @RightBorderLamps (Displacement)  SELECT L.Displacement  FROM #Lamps L  LEFT JOIN #Lamps L1 ON L1.Displacement - L.Displacement > 0 and L1.Displacement - L.Displacement < 50  WHERE L1.id\_ is NULL  ORDER BY 1  SELECT convert(numeric(10, 3), round(sum(R.Displacement - L.Displacement + 30)/1000.0, 3)) Len  INTO #LampsLength  FROM @LeftBorderLamps L  JOIN @RightBorderLamps R ON R.id\_ = L.id\_  SELECT 'Автопавильоны капитального типа, шт.',  (select convert(varchar(10), count(\*)) from ListBusStops where NumRoad = @NumRoad and NumDataSource = @NumDataSource),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT 'Площадки отдыха, шт.',  '-',  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT 'Площадки для стоянок и остановок автомобиля, шт.',  '-',  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT 'Освещение, км.',  (select convert(varchar(10), len) from #LampsLength),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT 'Переходно-скоростные полосы, шт.',  '-',  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT 'Ограждения, км.',  (select convert(varchar(10), ISNULL(convert(numeric(10,3), round(sum(EndPos - StartPos)/1000.0, 3)), 0)) from ListBarriers where NumRoad = @NumRoad and NumDataSource = @NumDataSource and NumTypeGuidingStructure <> 181),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT ' в том числе : барьерное',  (select convert(varchar(10), ISNULL(convert(numeric(10,3), round(sum(EndPos - StartPos)/1000.0, 3)), 0)) from ListBarriers where NumRoad = @NumRoad and NumDataSource = @NumDataSource and NumTypeGuidingStructure <> 181 and NumBarrierConstruction in (112, 113)),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT ' по типу "Нью-Джерси"',  (select convert(varchar(10), ISNULL(convert(numeric(10,3), round(sum(EndPos - StartPos)/1000.0, 3)), 0)) from ListBarriers where NumRoad = @NumRoad and NumDataSource = @NumDataSource and NumTypeGuidingStructure <> 181 and NumBarrierConstruction in (118)),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT ' перила',  (select convert(varchar(10), ISNULL(convert(numeric(10,3), round(sum(EndPos - StartPos)/1000.0, 3)), 0)) from ListBarriers where NumRoad = @NumRoad and NumDataSource = @NumDataSource and NumTypeGuidingStructure <> 181 and NumBarrierConstruction in (117)),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT ' прочие',  (select convert(varchar(10), ISNULL(convert(numeric(10,3), round(sum(EndPos - StartPos)/1000.0, 3)), 0)) from ListBarriers where NumRoad = @NumRoad and NumDataSource = @NumDataSource and NumTypeGuidingStructure <> 181 and ISNULL(NumBarrierConstruction, 0) not in (112,113,114,115,116,117,118,1959537)),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT 'Сигнальные столбики, шт.',  (select convert(varchar(10), ISNULL(sum(GuidingStructuresCount), 0)) from ListBarriers where NumRoad = @NumRoad and NumDataSource = @NumDataSource and NumTypeGuidingStructure = 181),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT 'Дорожные знаки, шт.',  (select convert(varchar(10), count(\*)) from ListRoadSigns where NumRoad = @NumRoad and NumDataSource = @NumDataSource),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT ' в том числе : предупреждающие',  (select convert(varchar(10), count(\*)) from ListRoadSigns S join Classifier C48 ON C48.id\_ = S.NumRoadSign where S.NumRoad = @NumRoad and S.NumDataSource = @NumDataSource and LEFT(C48.ShortTitle, 1) = '1'),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT ' приоритета',  (select convert(varchar(10), count(\*)) from ListRoadSigns S join Classifier C48 ON C48.id\_ = S.NumRoadSign where S.NumRoad = @NumRoad and S.NumDataSource = @NumDataSource and LEFT(C48.ShortTitle, 1) = '2'),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT ' запрещающие',  (select convert(varchar(10), count(\*)) from ListRoadSigns S join Classifier C48 ON C48.id\_ = S.NumRoadSign where S.NumRoad = @NumRoad and S.NumDataSource = @NumDataSource and LEFT(C48.ShortTitle, 1) = '3'),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT ' предписывающие',  (select convert(varchar(10), count(\*)) from ListRoadSigns S join Classifier C48 ON C48.id\_ = S.NumRoadSign where S.NumRoad = @NumRoad and S.NumDataSource = @NumDataSource and LEFT(C48.ShortTitle, 1) = '4'),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT ' особых предписаний',  (select convert(varchar(10), count(\*)) from ListRoadSigns S join Classifier C48 ON C48.id\_ = S.NumRoadSign where S.NumRoad = @NumRoad and S.NumDataSource = @NumDataSource and LEFT(C48.ShortTitle, 1) = '5'),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT ' информационные',  (select convert(varchar(10), count(\*)) from ListRoadSigns S join Classifier C48 ON C48.id\_ = S.NumRoadSign where S.NumRoad = @NumRoad and S.NumDataSource = @NumDataSource and LEFT(C48.ShortTitle, 1) = '6'),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT ' сервиса',  (select convert(varchar(10), count(\*)) from ListRoadSigns S join Classifier C48 ON C48.id\_ = S.NumRoadSign where S.NumRoad = @NumRoad and S.NumDataSource = @NumDataSource and LEFT(C48.ShortTitle, 1) = '7'),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT ' доп. информации',  (select convert(varchar(10), count(\*)) from ListRoadSigns S join Classifier C48 ON C48.id\_ = S.NumRoadSign where S.NumRoad = @NumRoad and S.NumDataSource = @NumDataSource and LEFT(C48.ShortTitle, 1) = '8'),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT 'Примыканий, шт.',  (select convert(varchar(10), count(\*)) from ListRoadAttachments where NumRoad = @NumRoad and NumDataSource = @NumDataSource),  '', '', '', '', '', '', '', '', '', ''    DROP TABLE #Lamps  DROP TABLE #LampsLength | | | | | | | | | | |

**4.10.3. СВОДНАЯ ВЕДОМОСТЬ НАЛИЧИЯ ТРУБ**

| **Виды сооружений** | **Наличие (шт./п.м) на 1 января** | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **2015г.** | **2016г.** | **2017г.** | **2018г.** | **2019г.** | **2020г.** | **2021г.** | **2022г.** | **2023г.** | **2024г.** |
| [CONTENT]  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*/3805/\*$\*/  SELECT @NumRoad = NumRoad FROM ListDataSources WHERE id\_ = @NumDataSource  SELECT \*  INTO #Tubes  FROM ListTubes T  WHERE T.NumRoad = @NumRoad  and T.NumDataSource = @NumDataSource  SELECT 'Металлические',  (SELECT ISNULL(convert(varchar(10), count(\*)) + '/' + convert(varchar(10), convert(int, round(sum(Extent), 0))), '') FROM #Tubes WHERE NumMaterial = 2292),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT 'Железобетонные',  (SELECT ISNULL(convert(varchar(10), count(\*)) + '/' + convert(varchar(10), convert(int, round(sum(Extent), 0))), '') FROM #Tubes WHERE NumMaterial = 2293),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT 'Деревянные',  '',  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT 'Прочие',  (SELECT ISNULL(convert(varchar(10), count(\*)) + '/' + convert(varchar(10), convert(int, round(sum(Extent), 0))), '') FROM #Tubes WHERE ISNULL(NumMaterial, 0) not in (2292, 2293)),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT ' итого:',  (SELECT ISNULL(convert(varchar(10), count(\*)) + '/' + convert(varchar(10), convert(int, round(sum(Extent), 0))), '') FROM #Tubes),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT 'в т.ч. на основной дороге:',  (SELECT ISNULL(convert(varchar(10), count(\*)) + '/' + convert(varchar(10), convert(int, round(sum(Extent), 0))), '') FROM #Tubes WHERE ISNULL(NumPlace, 0) not in (257, 258)),  '', '', '', '', '', '', '', '', '', ''  DROP TABLE #Tubes | | | | | | | | | | |

**4.10.7. СВОДНАЯ ВЕДОМОСТЬ ТРОТУАРОВ И ПЕШЕХОДНЫХ ДОРОЖЕК**

| **Наименование** | **Площадь (м2) на 1 января** | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **2015г.** | **2016г.** | **2017г.** | **2018г.** | **2019г.** | **2020г.** | **2021г.** | **2022г.** | **2023г.** | **2024г.** |
| [CONTENT]  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*/3805/\*$\*/  SELECT @NumRoad = NumRoad FROM ListDataSources WHERE id\_ = @NumDataSource  SELECT 'Тротуары',  convert(varchar(10), convert(int, round(sum((EndPos - StartPos) \* Width), 0))),  '', '', '', '', '', '', '', '', '', ''  FROM ListFootWay  WHERE NumRoad = @NumRoad  and NumDataSource = @NumDataSource | | | | | | | | | | |

**4.10.8. СВОДНАЯ ВЕДОМОСТЬ УКРЕПЛЕНИЯ ОБОЧИН**

| **Наименование** | **Протяженность (км) на 1 января** | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **2015г.** | **2016г.** | **2017г.** | **2018г.** | **2019г.** | **2020г.** | **2021г.** | **2022г.** | **2023г.** | **2024г.** |
| [CONTENT]  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*/3805/\*$\*/  SELECT @NumRoad = NumRoad FROM ListDataSources WHERE id\_ = @NumDataSource  SELECT C.NumTypeStrengthening, convert(numeric(10,3), round(sum((S.EndPos - S.StartPos))/1000.0, 3)) len  INTO #Sides  FROM ListRoadSides S  JOIN CharactRoadSides C ON C.NumRoadSide = S.id\_  WHERE S.NumRoad = @NumRoad  and S.NumDataSource = @NumDataSource  GROUP BY C.NumTypeStrengthening  SELECT 'Укрепленные обочины, всего',  (select ISNULL(convert(varchar(10), sum(len)), '-') from #Sides),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT 'В том числе укреплённые: щебнем (гравием)',  (select ISNULL(convert(varchar(10), sum(len)), '-') from #Sides where NumTypeStrengthening = 94),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT ' бетоном',  (select ISNULL(convert(varchar(10), sum(len)), '-') from #Sides where NumTypeStrengthening = 90),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT ' асфальтобетоном',  (select ISNULL(convert(varchar(10), sum(len)), '-') from #Sides where NumTypeStrengthening = 88),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT ' прочими материалами',  (select ISNULL(convert(varchar(10), sum(len)), '-') from #Sides where ISNULL(NumTypeStrengthening, 0) not in (88, 90, 94)),  '', '', '', '', '', '', '', '', '', ''  DROP TABLE #Sides | | | | | | | | | | |

**4.10.9. СВОДНАЯ ВЕДОМОСТЬ СЪЕЗДОВ (ВЪЕЗДОВ)**

| **Тип покрытия съезда**  **(въезда)** | **Количество** | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **2015г.** | **2016г.** | **2017г.** | **2018г.** | **2019г.** | **2020г.** | **2021г.** | **2022г.** | **2023г.** | **2024г.** |
| [CONTENT]  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*/3805/\*$\*/  SELECT @NumRoad = NumRoad FROM ListDataSources WHERE id\_ = @NumDataSource  SELECT \*  INTO #Attachments  FROM ListRoadAttachments  WHERE NumRoad = @NumRoad  and NumDataSource = @NumDataSource    SELECT 'Усовершенствованное',  (select count(\*) from #Attachments where NumTech = 193),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT 'Переходное',  (select count(\*) from #Attachments where NumTech = 194),  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT 'Прочее',  '0',  '', '', '', '', '', '', '', '', '', '' UNION ALL  SELECT 'Всего',  (select count(\*) from #Attachments),  '', '', '', '', '', '', '', '', '', ''    DROP TABLE #Attachments | | | | | | | | | | |

***ВЕДОМОСТЬ***

***соответствия проектного километража эксплуатационному на автомобильной дороге***

| Местоположение проектного километража, км+ | Эксплуатационный километраж | | Расстояние между эксплуатационными километровыми столбами | Тип существования | Примечание |
| --- | --- | --- | --- | --- | --- |
| Информация прямого направления километрового знака | Информация обратного направления километрового знака |
| **[CONTENT]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*/3805/\*$\*/  SELECT @NumRoad = NumRoad FROM ListDataSources WHERE id\_ = @NumDataSource  DECLARE @RoadStart int, @RoadEnd int  SELECT @RoadStart = round(StartPos, 0), @RoadEnd = round(EndPos, 0)  FROM ListRoadParts  WHERE NumPartType = 33  and NumRoad = @NumRoad  and NumDataSource = @NumDataSource  SELECT id\_, Displacement, SignLabel  INTO #Signs  FROM ListRoadSigns S  WHERE S.NumRoad = @NumRoad  and S.NumDataSource = @NumDataSource  and S.NumRoadSign = 43752  SELECT dbo.pp\_Km\_mFormatTP(S.Displacement) [Адрес],  LEFT(S.SignLabel, ISNULL(NULLIF(CHARINDEX('-', S.SignLabel), 0) - 1, LEN(S.SignLabel))) [Надпись прямое],  RIGHT(rtrim(S.SignLabel), NULLIF(LEN(S.SignLabel) - CHARINDEX('-', S.SignLabel), 0)) [Надпись обратное],  ISNULL(convert(varchar(10), convert(int, S.Displacement - (select max(Displacement) from #Signs where Displacement < S.Displacement))), '') [Расстояние],  'Существующий' [Тип существования]  FROM #Signs S  ORDER BY S.Displacement  DROP TABLE #Signs | | | | | |

***ВЕДОМОСТЬ***

***наличия и технического состояния мостов (путепроводов) на автомобильной дороге***

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Местоположение, км + | Наименование сооружения | Наименование перекрываемого препятствия | Общая длина | Длина подходов | Год | | Техническая характеристика сооружения | | | | | Техническое состояние |
| Постройки | Последнего кап. ремонта | материал пролетных сооружений | тип пролетных строений | габарит проезда | нормативная нагрузка | грузоподъемность, т (общая/осевая/одиночная) |
|  |  |  |  |  |  |  |  |  |  |  |  |  |

***ВЕДОМОСТЬ***

***наличия и технического состояния водопропускных труб на автомобильной дороге***

| Местоположение | | Тип водотока | Форма поперечного сечения | Кол-во очков | Материал | Диаметр (ширина), м | Высота, м | Полная длина, м | Объем тела трубы, м | Техниеское состояние |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| км | расположение |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ВСЕГО КОЛИЧЕСТВО ТРУБ | | : |  | ОБЩЕЙ ДЛИНОЙ | : |  | ОБЩИМ ОБЪЕМОМ | : |  |
| В том числе : | Металлических | : |  | общей длиной | : |  | общим объемом | : |  |
| Железобетонных | : |  | общей длиной | : |  | общим объемом | : |  |
| Бетонных | : | 0 | общей длиной | : | 0 | общим объемом | : | 0 |
| Каменных | : | 0 | общей длиной | : | 0 | общим объемом | : | 0 |
| Деревянных | : | 0 | общей длиной | : | 0 | общим объемом | : | 0 |
| Асбоцементных | : | 0 | общей длиной | : | 0 | общим объемом | : | 0 |
| Прочих | : | 0 | общей длиной | : | 0 | общим объемом | : | 0 |
| В ТОМ ЧИСЛЕ ПО ОСНОВНОЙ ДОРОГЕ | |  |  |  |  |  |  |  |  |
| ВСЕГО КОЛИЧЕСТВО ТРУБ | | : |  | ОБЩЕЙ ДЛИНОЙ | : |  | ОБЩИМ ОБЪЕМОМ | : |  |
| В том числе : | Металлических | : |  | общей длиной | : |  | общим объемом | : |  |
| Железобетонных | : |  | общей длиной | : |  | общим объемом | : |  |
| Бетонных | : | 0 | общей длиной | : | 0 | общим объемом | : | 0 |
| Каменных | : | 0 | общей длиной | : | 0 | общим объемом | : | 0 |
| Деревянных | : | 0 | общей длиной | : | 0 | общим объемом | : | 0 |
| Асбоцементных | : | 0 | общей длиной | : | 0 | общим объемом | : | 0 |
| Прочих | : | 0 | общей длиной | : | 0 | общим объемом | : | 0 |

***ВЕДОМОСТЬ***

***наличия и технического состояния автобусных остановок на автомобильной дороге***

| Местоположение, км+ | | Название | Наличие элементов | | | Материал павильона | Техническое состояние | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Слева | Справа | остановочная площадка | переходно-скоростная полоса | посадочная площадка | остановочной площадки | посадочной площадки | павильона |
| **[CONTENT]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*/3805/\*$\*/  SELECT @NumRoad = NumRoad  FROM ListDataSources  WHERE id\_ = @NumDataSource  SELECT  case when NumPlace = 257 then dbo.pp\_Km\_mFormatTP(T.Displacement) else '' end СЛЕВА,  case when NumPlace = 258 then dbo.pp\_Km\_mFormatTP(T.Displacement) else '' end СПРАВА,  '' Название,  T.StopArea [остановочная площадка],  '-' ПСП,  T.EmbarkationArea [посадочная площадка],  rtrim(Cpl.FullTitle) [расположение],  T.Pavilion [павильон],  '', '', T.Cond [состояние]  FROM (  SELECT B.id\_, B.Displacement, B.NumPlace,  (case when B.Pavilion = 1 then ISNULL(rtrim(M.FullTitle), 'неизв.') else '-' end) Pavilion,  (case when B.EmbarkationArea = 1 then 'есть' else '-' end) EmbarkationArea,  (case when B.StopArea = 1 then 'есть' else '-' end) StopArea,  case when B.IsDef\_PavilionTooClose = 1 then 'V' else '' end [Близкое расположение павильона],  case when B.IsDef\_CurveTooSmall = 1 then 'V' else '' end [Расположение на кривой менее допустимого],  case when B.IsDef\_SlopeTooBig = 1 then 'V' else '' end [Расположение на уклоне более допустимого],  case when B.IsDef\_VisibilityTooSmall = 1 then 'V' else '' end [Недостаточное расстояние видимости],  case when B.IsDef\_PavilionsWrongLocation = 1 then 'V' else '' end [Недостаточное расстояние (неправильное расположение) между павильонами],  case when B.IsDef\_Other = 1 then 'V' else '' end [Прочее],  ISNULL(rtrim(CC.ShortTitle), 'удовл.') Cond  FROM ListBusStops B  LEFT JOIN Classifier CC ON CC.id\_ = B.NumCondition  LEFT JOIN ListMaterials M ON M.id\_ = B.NumMaterial  WHERE B.NumRoad = @NumRoad  and B.NumDataSource = @NumDataSource  UNION  SELECT -S.id\_, S.Displacement, (case when S.NumMoveDirection = 246 then 258 else 257 end) NumPlace,  '-' [павильон],  '-' EmbarkationArea,  '-' StopArea,  '-' [Близкое расположение павильона],  '-' [Расположение на кривой менее допустимого],  '-' [Расположение на уклоне более допустимого],  '-' [Недостаточное расстояние видимости],  '-' [Недостаточное расстояние (неправильное расположение) между павильонами],  '-' [Прочее],  '-'  FROM ListRoadSigns S  LEFT JOIN ListBusStops B ON B.NumDataSource = @NumDataSource  and B.NumRoad = @NumRoad  and B.NumPlace = (case when S.NumMoveDirection = 246 then 258 else 257 end)  and B.Displacement between S.Displacement - 50 and S.Displacement + 50  WHERE S.NumRoad = @NumRoad  and S.NumDataSource = @NumDataSource  and S.NumPlace = 258  and S.NumRoadSign = 43726  and B.id\_ is NULL  ) T  LEFT JOIN Classifier Cpl ON Cpl.id\_ = T.NumPlace  ORDER BY T.Displacement | | | | | | | | | |

***ВЕДОМОСТЬ***

***наличия и технического состояния переходно-скоростных на автомобильной дороге***

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Местоположение, км+ | | Протяженность, км | | Площадь покрытия, кв.м | | | Тип покрытия | Год | | Техническое состояние |
| Начало | Конец | Слева | Справа | Всего | В том числе | | постройки | последнего кап. ремонта |
| полоса торможения | полоса разгона |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |

***ВЕДОМОСТЬ***

***наличия и технического состояния уширений на автомобильной дороге***

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Местоположение, км+ | | Тип уширения | Протяженность | | Площадь покрытия, кв.м | | | Тип покрытия | Принадлежность к объекту | Год | | Техническое состояние |
| Начало | Конец | Слева | Справа | Всего | В том числе | | Постройки | Последнего кап.ремонта |
| Полоса торможения | Полоса разгона |
|  |  |  |  |  |  |  |  |  |  |  |  |  |

***ВЕДОМОСТЬ***

***наличия и технического состояния дорожных знаков на автомобильной дороге***

| № п/п | Наименование дорожного знака | Номер знака по ГОСТу | Местоположение, км+ | | | Обслуживаемое направление | Примечание | Техническое состояние |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| слева | спрва | раздел. полоса |
| **[CONTENT]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*/3805/\*$\*/  SELECT @NumRoad = NumRoad  FROM ListDataSources  WHERE id\_ = @NumDataSource  DECLARE @RoadStart int, @RoadEnd int  SELECT @RoadStart = round(StartPos, 0), @RoadEnd = round(EndPos, 0)  FROM ListRoadParts  WHERE NumPartType = 33  and NumRoad = @NumRoad  and NumDataSource = @NumDataSource  CREATE TABLE #Result (id\_ int identity(1,1), SignName varchar(200), SignCode varchar(100),  OnLeft varchar(100), OnRight varchar(100), OnCenter varchar(100),  Direction varchar(100), Comment varchar(200), T varchar(100))  INSERT INTO #Result (SignName, SignCode,  OnLeft, OnRight, OnCenter,  Direction, Comment, T)  SELECT rtrim(C.FullTitle),  rtrim(C.ShortTitle),  (case when (S.NumPlace = 257 and S.NumMoveDirection = 246)  or (S.NumPlace = 258 and S.NumMoveDirection = 247) then dbo.pp\_Km\_mFormatTP(S.Displacement) else '' end) [Слева],  (case when (S.NumPlace = 258 and S.NumMoveDirection = 246)  or (S.NumPlace = 257 and S.NumMoveDirection = 247) then dbo.pp\_Km\_mFormatTP(S.Displacement) else '' end) [Справа],  (case S.NumPlace when 259 then dbo.pp\_Km\_mFormatTP(S.Displacement) else '' end) [На разделительной полосе],  rtrim(CD.FullTitle),  rtrim(S.SignLabel),  'удовл.'  FROM ListRoadSigns S  JOIN Classifier C ON C.id\_ = S.NumRoadSign  JOIN Classifier CD ON CD.id\_ = S.NumMoveDirection  WHERE S.NumRoad = @NumRoad  and S.NumDataSource = @NumDataSource  and Displacement between @RoadStart and @RoadEnd  ORDER BY S.Displacement  SELECT id\_ [№ п/п],  SignName [Наименование знака],  SignCode [Номер знака по ГОСТу],  OnLeft [Слева],  OnRight [Справа],  OnCenter [На разделительной полосе],  Direction [Обслуживаемое направление],  Comment [Примечание],  T [Техническое состояние]  FROM #Result  ORDER BY id\_  DROP TABLE #Result | | | | | | | | |

|  |
| --- |
| **[CONTENT][AUTO\_EXPAND]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*/3805/\*$\*/  SELECT @NumRoad = NumRoad  FROM ListDataSources  WHERE id\_ = @NumDataSource  DECLARE @RoadStart int, @RoadEnd int  SELECT @RoadStart = round(StartPos, 0), @RoadEnd = round(EndPos, 0)  FROM ListRoadParts  WHERE NumPartType = 33  and NumRoad = @NumRoad  and NumDataSource = @NumDataSource  CREATE TABLE #Result (id\_ int identity(1,1), SignName varchar(200), SignCode varchar(100),  OnLeft varchar(100), OnRight varchar(100), OnCenter varchar(100),  Direction varchar(100), Comment varchar(200), T varchar(100))  INSERT INTO #Result (SignName, SignCode,  OnLeft, OnRight, OnCenter,  Direction, Comment, T)  SELECT rtrim(C.FullTitle),  rtrim(C.ShortTitle),  (case when (S.NumPlace = 257 and S.NumMoveDirection = 246)  or (S.NumPlace = 258 and S.NumMoveDirection = 247) then dbo.pp\_Km\_mFormatTP(S.Displacement) else '' end) [Слева],  (case when (S.NumPlace = 258 and S.NumMoveDirection = 246)  or (S.NumPlace = 257 and S.NumMoveDirection = 247) then dbo.pp\_Km\_mFormatTP(S.Displacement) else '' end) [Справа],  (case S.NumPlace when 259 then dbo.pp\_Km\_mFormatTP(S.Displacement) else '' end) [На разделительной полосе],  rtrim(CD.FullTitle),  rtrim(S.SignLabel),  'удовл.'  FROM ListRoadSigns S  JOIN Classifier C ON C.id\_ = S.NumRoadSign  JOIN Classifier CD ON CD.id\_ = S.NumMoveDirection  WHERE S.NumRoad = @NumRoad  and S.NumDataSource = @NumDataSource  and Displacement between @RoadStart and @RoadEnd  ORDER BY S.Displacement  SELECT DISTINCT LEFT(SignCode, 1) Code,  (case LEFT(SignCode, 1)  when '1' then 'Предупреждающие'  when '2' then 'Знаки приоритета'  when '3' then 'Запрещающие'  when '4' then 'Предписывающие'  when '5' then 'Знаки особых предписаний'  when '6' then 'Информационные'  when '7' then 'Знаки сервиса'  when '8' then 'Знаки доп. информации'  else '???'  end) CodeName  INTO #SignTypes  FROM #Result  SELECT CodeName, NumSigns  FROM  (  SELECT 0 Code, 'ВСЕГО' CodeName, count(\*) NumSigns  FROM #Result  UNION  SELECT T.Code, T.CodeName, count(\*) NumSigns  FROM #SignTypes T  JOIN #Result R ON LEFT(R.SignCode, 1) = T.Code  GROUP BY T.Code, T.CodeName  ) T  ORDER BY T.Code  DROP TABLE #Result  DROP TABLE #SignTypes |

***ВЕДОМОСТЬ***

***наличия и технического состояния обочин слева на автомобильной дороге***

| Месторасположение, км+ | | Протяженность, км | Ширина, м | Площадь, кв.м | Материал | Техническое состояние |
| --- | --- | --- | --- | --- | --- | --- |
| начало | конец |
| **[CONTENT]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*/3805/\*$\*/  SELECT @NumRoad = NumRoad  FROM ListDataSources  WHERE id\_ = @NumDataSource  SELECT S.StartPos, S.EndPos, S.NumPlace,  (S.EndPos - S.StartPos)/1000.0 Length, C.RowWidth Width,  rtrim(cs.FullTitle) Surf,  'удовл.' T  INTO #Result  FROM ListRoadSides S  JOIN CharactRoadSides C ON C.NumRoadSide = S.id\_  JOIN Classifier CS ON CS.id\_ = C.NumTypeStrengthening  WHERE S.NumRoad = @NumRoad  and S.NumDataSource = @NumDataSource  ORDER BY S.StartPos  SELECT dbo.pp\_Km\_mFormatTP(StartPos) [Начало], dbo.pp\_Km\_mFormatTP(EndPos) [Конец],  convert(decimal(10,3), Length) [Протяженность, км], convert(decimal(10, 1), Width) [Ширина, м],  convert(decimal(10,2), Length\*1000\*Width) [Площадь, м2],  Surf [Матариал],  T [Техническое состояние]  FROM #Result  WHERE NumPlace = 257  ORDER BY StartPos  DROP TABLE #Result | | | | | | |

|  |
| --- |
| **[CONTENT][AUTO\_EXPAND]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*/3805/\*$\*/  SELECT @NumRoad = NumRoad  FROM ListDataSources  WHERE id\_ = @NumDataSource  SELECT S.StartPos, S.EndPos, S.NumPlace,  (S.EndPos - S.StartPos)/1000.0 Length, C.RowWidth Width,  rtrim(cs.FullTitle) Surf,  'удовл.' T  INTO #Result  FROM ListRoadSides S  JOIN CharactRoadSides C ON C.NumRoadSide = S.id\_  JOIN Classifier CS ON CS.id\_ = C.NumTypeStrengthening  WHERE S.NumRoad = @NumRoad  and S.NumDataSource = @NumDataSource  ORDER BY S.StartPos  SELECT Surf [Матариал],  sum(convert(decimal(10,2), Length\*1000\*Width)) [Площадь, м2]  FROM #Result  WHERE NumPlace = 257  GROUP BY Surf  DROP TABLE #Result |

***ВЕДОМОСТЬ***

***наличия и технического состояния обочин справа на автомобильной дороге***

| Месторасположение, км+ | | Протяженность, км | Ширина, м | *Площадь, кв.м* | Материал | Техническое состояние |
| --- | --- | --- | --- | --- | --- | --- |
| начало | конец |
| ***[CONTENT]***  *DECLARE @NumRoad int, @NumDataSource int*  *SELECT @NumDataSource=/\*$NumDataSource\*/3805/\*$\*/*  *SELECT @NumRoad = NumRoad*  *FROM ListDataSources*  *WHERE id\_ = @NumDataSource*  *SELECT S.StartPos, S.EndPos, S.NumPlace,*  *(S.EndPos - S.StartPos)/1000.0 Length, C.RowWidth Width,*  *rtrim(cs.FullTitle) Surf,*  *'удовл.' T*  *INTO #Result*  *FROM ListRoadSides S*  *JOIN CharactRoadSides C ON C.NumRoadSide = S.id\_*  *JOIN Classifier CS ON CS.id\_ = C.NumTypeStrengthening*  *WHERE S.NumRoad = @NumRoad*  *and S.NumDataSource = @NumDataSource*  *ORDER BY S.StartPos*  *SELECT dbo.pp\_Km\_mFormatTP(StartPos) [Начало], dbo.pp\_Km\_mFormatTP(EndPos) [Конец],*  *convert(decimal(10,3), Length) [Протяженность, км], convert(decimal(10, 1), Width) [Ширина, м],*  *convert(decimal(10,2), Length\*1000\*Width) [Площадь, м2],*  *Surf [Матариал],*  *T [Техническое состояние]*  *FROM #Result*  *WHERE NumPlace = 258*  *ORDER BY StartPos*  *DROP TABLE #Result* | | | | | | |

|  |
| --- |
| **[CONTENT][AUTO\_EXPAND]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*/3805/\*$\*/  SELECT @NumRoad = NumRoad  FROM ListDataSources  WHERE id\_ = @NumDataSource  SELECT S.StartPos, S.EndPos, S.NumPlace,  (S.EndPos - S.StartPos)/1000.0 Length, C.RowWidth Width,  rtrim(cs.FullTitle) Surf,  'удовл.' T  INTO #Result  FROM ListRoadSides S  JOIN CharactRoadSides C ON C.NumRoadSide = S.id\_  JOIN Classifier CS ON CS.id\_ = C.NumTypeStrengthening  WHERE S.NumRoad = @NumRoad  and S.NumDataSource = @NumDataSource  ORDER BY S.StartPos  SELECT Surf [Матариал],  sum(convert(decimal(10,2), Length\*1000\*Width)) [Площадь, м2]  FROM #Result  WHERE NumPlace = 258  GROUP BY Surf  DROP TABLE #Result |

***ВЕДОМОСТЬ***

***наличия коммуникаций, находящихся в пределах полосы отвода и придорожной полосы на автомобильной дороге***

| Местоположение, км+ | | Местоположение относительно оси дороги | Тип | Высота над проезжей частью | Кол-во проводов | Расстояние от кромки проезжей части | | Владелец |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Начало | Конец | Слева | Справа |
| **[CONTENT]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*/3805/\*$\*/  SELECT @NumRoad = NumRoad  FROM ListDataSources  WHERE id\_ = @NumDataSource  DECLARE @RoadStart int, @RoadEnd int  SELECT @RoadStart = round(StartPos, 0), @RoadEnd = round(EndPos, 0)  FROM ListRoadParts  WHERE NumPartType = 33  and NumRoad = @NumRoad  and NumDataSource = @NumDataSource  SELECT dbo.pp\_Km\_mFormatTP(S.StartPos) [Начало], dbo.pp\_Km\_mFormatTP(S.EndPos) [Конец],  rtrim(CP.Fulltitle) [Местоположение],  rtrim(CT.FullTitle) [Наименование],  '' [Высота над проезжей частью],  '' [Количество проводов],  '' [Расстояние от кромки]  FROM ListCommunicationStruct S  LEFT JOIN Classifier CP ON CP.id\_ = S.NumPlaceRoad  LEFT JOIN Classifier CT ON CT.id\_ = S.NumTypeCOmStructure  WHERE S.NumRoad = @NumRoad  and S.NumDataSource = @NumDataSource  and S.StartPos between @RoadStart and @RoadEnd  ORDER BY S.StartPos | | | | | | | | |

***ВЕДОМОСТЬ***

***размеров полосы отвода на автомобильной дороге***

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Участок | | | Размеры полосы отвода |  | Наименование, номер и дата документа отвода земель | Сведения о натуральном закреплении границ полосы отвода |
| От км+ | До км+ | Протяженность, км | Слева от оси дороги | Справа от оси дороги |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

***ВЕДОМОСТЬ***

***наличия и технического состояния водоотводных сооружений на автомобильной дороге***

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Местоположение, км+ | Протяженность, км | | | Тип устройства | Материал конструкции | Характеристики | | Год | | Техническое состояние |
| Начало | Конец | Слева | Справа | Ширина, м | Высота, м | Постройки | Посл. кап. ремонта |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |

***ВЕДОМОСТЬ***

***наличия и технического состояния примыканий и пересечений в одном уровне на автомобильной дороге***

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Местоположение | | Направление | Угол примыкания | Тип покрытия | Обустройство | | | |
| Км+ | Расположение | Переходно-скоростные полосы | в/п труба | Островок безопасности | Сигнальные столбики |
| **[CONTENT]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*/3805/\*$\*/  SELECT @NumRoad = NumRoad  FROM ListDataSources  WHERE id\_ = @NumDataSource  DECLARE @RoadStart int, @RoadEnd int  SELECT @RoadStart = round(StartPos, 0), @RoadEnd = round(EndPos, 0)  FROM ListRoadParts  WHERE NumPartType = 33  and NumRoad = @NumRoad  and NumDataSource = @NumDataSource  SELECT S.id\_, sum(B.GuidingStructuresCount) NumCount  INTO #AttSignalColumns  FROM ListRoadAttachments S  JOIN ListBarriers B ON B.NumRoad = S.NumRoad  and B.NumDataSource = S.NumDataSource  and B.NumTypeGuidingStructure = 181  and S.StartPos between B.StartPos - 40 and B.StartPos + 40  WHERE S.NumRoad = @NumRoad  and S.NumDataSource = @NumDataSource  and S.StartPos between @RoadStart and @RoadEnd  GROUP BY S.id\_  SELECT DISTINCT S.id\_  INTO #AttTubes  FROM ListRoadAttachments S  JOIN ListTubes B ON B.NumRoad = S.NumRoad  and B.NumDataSource = S.NumDataSource  and B.NumPlace = S.NumPlace  and S.StartPos between B.Displacement - 40 and B.Displacement + 40  WHERE S.NumRoad = @NumRoad  and S.NumDataSource = @NumDataSource  and S.StartPos between @RoadStart and @RoadEnd  SELECT DISTINCT S.id\_  INTO #AttInha  FROM ListRoadAttachments S  JOIN ListRoadParts B ON B.NumRoad = S.NumRoad  and B.NumDataSource = S.NumDataSource  and B.NumPartType = 9  and S.StartPos between B.StartPos and B.EndPos  WHERE S.NumRoad = @NumRoad  and S.NumDataSource = @NumDataSource  and S.StartPos between @RoadStart and @RoadEnd  SELECT dbo.pp\_Km\_mFormatTP(S.StartPos) [Адрес],  rtrim(CP.FullTitle) [Расположение],  rtrim(convert(varchar(100), ISNULL(NULLIF(S.AttachmentName, ''), (case when I.id\_ is not NULL then 'к домам' else 'в поле' end)))) [Наименование],  convert(int, S.CrossingCorner) [Угол],  (case S.NumTech when 193 then 'асфальтобетонное' else 'грунтовое' end) [Тип покрытия],  'Нет' [Переходно-скорокстные полосы],  (case when T.id\_ is NULL then 'Нет' else 'Да' end) [В/п труба],  'Нет' [Островок безопасности],  ISNULL(convert(varchar(10), (SC.NumCount)), '') [Сигнальные столбики]  FROM ListRoadAttachments S  LEFT JOIN Classifier CP ON CP.id\_ = S.NumPlace  LEFT JOIN #AttSignalColumns SC ON SC.id\_ = S.id\_  LEFT JOIN #AttTubes T ON T.id\_ = S.id\_  LEFT JOIN #AttInha I ON I.id\_ = S.id\_  WHERE S.NumRoad = @NumRoad  and S.NumDataSource = @NumDataSource  and S.StartPos between @RoadStart and @RoadEnd  ORDER BY S.StartPos  DROP TABLE #AttSIgnalColumns  DROP TABLE #AttTubes  DROP TABLE #AttInha | | | | | | | | |

***ВЕДОМОСТЬ***

***прохождения по населенным пунктам автомобильной дороги***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Наименование населенного пункта | Местоположение, км+ | | Протяжение, м | Застройка | | | |
| Начало | Конец | Местоположение | | | Расстояние, м |
| Начало, км+ | Конец, км+ | Расположение |
| **[CONTENT]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*/3805/\*$\*/  SELECT @NumRoad = NumRoad  FROM ListDataSources  WHERE id\_ = @NumDataSource  DECLARE @RoadStart int, @RoadEnd int  SELECT @RoadStart = round(StartPos, 0), @RoadEnd = round(EndPos, 0)  FROM ListRoadParts  WHERE NumPartType = 33  and NumRoad = @NumRoad  and NumDataSource = @NumDataSource  SELECT rtrim(I.FullTitle) [Наименование],  dbo.pp\_Km\_mFormatTP(S.StartPos) [Начало], dbo.pp\_Km\_mFormatTP(S.EndPos) [Конец],  convert(int, S.EndPos - S.StartPos) [Протяженность],  dbo.pp\_Km\_mFormatTP(MP.StartPos) [Застройка Начало],  dbo.pp\_Km\_mFormatTP(MP.EndPos) [Застройка Конец],  rtrim(CP.FullTitle) [Застройка Положение],  ISNULL(convert(varchar(10), convert(int, MP.MinLength)), '') [Расстояние]  FROM ListRoadParts S  LEFT JOIN ListInhabitedLocalities I ON I.id\_ = S.Reference  LEFT JOIN ELEM\_MapObjects MP ON MP.NumRoad = S.NumRoad  and MP.NumDataSource = S.NumDataSource  and MP.StartPos between S.StartPos and S.EndPos  LEFT JOIN Classifier CP ON CP.id\_ = MP.NumPlace  WHERE S.NumRoad = @NumRoad  and S.NumDataSource = @NumDataSource  and S.StartPos between @RoadStart and @RoadEnd  and S.NumPartType = 9  ORDER BY S.StartPos, MP.StartPos | | | | | | | |

***ВЕДОМОСТЬ***

***наличия и технического состояния площадок стоянки и остановки автомобилей, отдыха водителей и пассажиров на автомобильной дороге***

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Местоположение, км+ | | Протяженность, км | | Площадь покрытия, кв.м | Тип покрытия | Год | | Техническое состояние |
| Начало | Конец | Слева | Справа | постройки | посл. кап. ремонта |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |

***ВЕДОМОСТЬ***

***наличия и технического состояния ограждений на автомобильной дороге***

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Местоположение, км+ | | | Протяженность, м | | | Тип | Материал | Год установки | Техническое состояние |
| Начало | Конец | Расположение | Слева | Справа | Раздел. полоса |
| **[CONTENT]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*/3805/\*$\*/  SELECT @NumRoad = NumRoad  FROM ListDataSources  WHERE id\_ = @NumDataSource  SELECT dbo.pp\_Km\_mFormatTP(B.StartPos) [Начало], dbo.pp\_Km\_mFormatTP(B.EndPos) [Конец],  '' [Расположение],  convert(int, (case when B.NumPlace = 257 then B.EndPos - B.StartPos else 0 end)) [Слева, м],  convert(int, (case when B.NumPlace = 258 then B.EndPos - B.StartPos else 0 end)) [Справа, м],  0 [Раздел полоса, м],  rtrim(CBc.FullTitle),  rtrim(CMt.FullTitle),  '' [год установки],  rtrim(ISNULL(CC.ShortTitle, 'удовл.'))  FROM ListBarriers B  JOIN dbo.Classifier CPl ON Cpl.id\_ = B.NumPlace  LEFT JOIN dbo.Classifier CBc ON CBc.id\_ = B.NumBarrierConstruction  LEFT JOIN dbo.ListMaterials CMt ON CMt.id\_ = B.NumMaterial  LEFT JOIN dbo.Classifier CC ON CC.id\_ = NULLIF(B.NumCondition, 0)  WHERE B.NumRoad = @NumRoad  and B.NumDataSource = @NumDataSource  and B.NumTypeGuidingStructure = 0  ORDER BY B.StartPos | | | | | | | | | |

***ВЕДОМОСТЬ***

***наличия и технического состояния направляющих устройств на автомобильной дороге***

| Наименование | Местоположение | | | Количество | | | Материал | Год установки | Техническое состояние | Зона расположения |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Начало, км+ | Конец, км+ | Расположение | Слева | Справа | Раздел. полоса |
| **[CONTENT]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*/3805/\*$\*/  SELECT @NumRoad = NumRoad  FROM ListDataSources  WHERE id\_ = @NumDataSource  DECLARE @RoadStart int, @RoadEnd int  SELECT @RoadStart = round(StartPos, 0), @RoadEnd = round(EndPos, 0)  FROM ListRoadParts  WHERE NumPartType = 33  and NumRoad = @NumRoad  and NumDataSource = @NumDataSource  SELECT B.\*  INTO #SignalColumns  FROM ListBarriers B  WHERE B.NumRoad = @NumRoad  and B.NumDataSource = @NumDataSource  and B.NumTypeGuidingStructure = 181  SELECT DISTINCT S.id\_, LRA.id\_ AID, convert(varchar(100), 'Примыкание ' + lower(rtrim(CP.FullTitle))) Zone  INTO #att  FROM #SignalColumns S  JOIN ListRoadAttachments LRA ON LRA.NumRoad = S.NumRoad  and LRA.NumDataSource = S.NumDataSource  and LRA.NumPlace = S.NumPlace  and LRA.StartPos between S.StartPos - 40 and S.StartPos + 40  JOIN Classifier CP ON CP.id\_ = LRA.NumPlace  SELECT DISTINCT S.id\_, LT.id\_ TID, convert(varchar(100), 'Труба') Zone  INTO #tubes  FROM #SignalColumns S  JOIN ListTubes LT ON LT.NumRoad = S.NumRoad  and LT.NumDataSource = S.NumDataSource  and LT.NumPlace = 262  and LT.Displacement between S.StartPos - 40 and S.StartPos + 40  SELECT min(B.StartPos) StartPos, max(B.EndPos) EndPos,  sum(case when B.NumPlace = 257 then B.GuidingStructuresCount else NULL end) LeftCount,  sum(case when B.NumPlace = 258 then B.GuidingStructuresCount else NULL end) RightCount,  ISNULL(t.TID, ISNULL(att.AID, B.StartPos)) GroupColumn,  ISNULL(t.Zone, ISNULL(att.Zone, 'Насыпь')) Zone  INTO #GroupColumns  FROM #SignalColumns B  LEFT JOIN #att att ON att.id\_ = B.id\_  LEFT JOIN #tubes t ON t.id\_ = B.id\_  GROUP BY ISNULL(t.TID, ISNULL(att.AID, B.StartPos)),  ISNULL(t.Zone, ISNULL(att.Zone, 'Насыпь'))  SELECT 'Сигнальные столбики' [Наименование], dbo.pp\_Km\_mFormatTP(B.StartPos) [Начало], dbo.pp\_Km\_mFormatTP(B.EndPos) [Конец],  (case when B.Zone like 'Примыкание%' then 'Примыкание' else 'Основная дорога' end) [Расположение],  ISNULL(convert(varchar(10), B.LeftCount), '') [Слева],  ISNULL(convert(varchar(10), B.RightCount), '') [Справа],  '' [Раздел. полоса],  'ж/б' [Материал],  '' [Год установки],  'удовл.' [Тех.состояние],  B.Zone [Зона расположения]  FROM #GroupColumns B  order by B.StartPos  DROP TABLE #SignalColumns  DROP TABLE #att  DROP TABLE #tubes  DROP TABLE #GroupColumns | | | | | | | | | | |